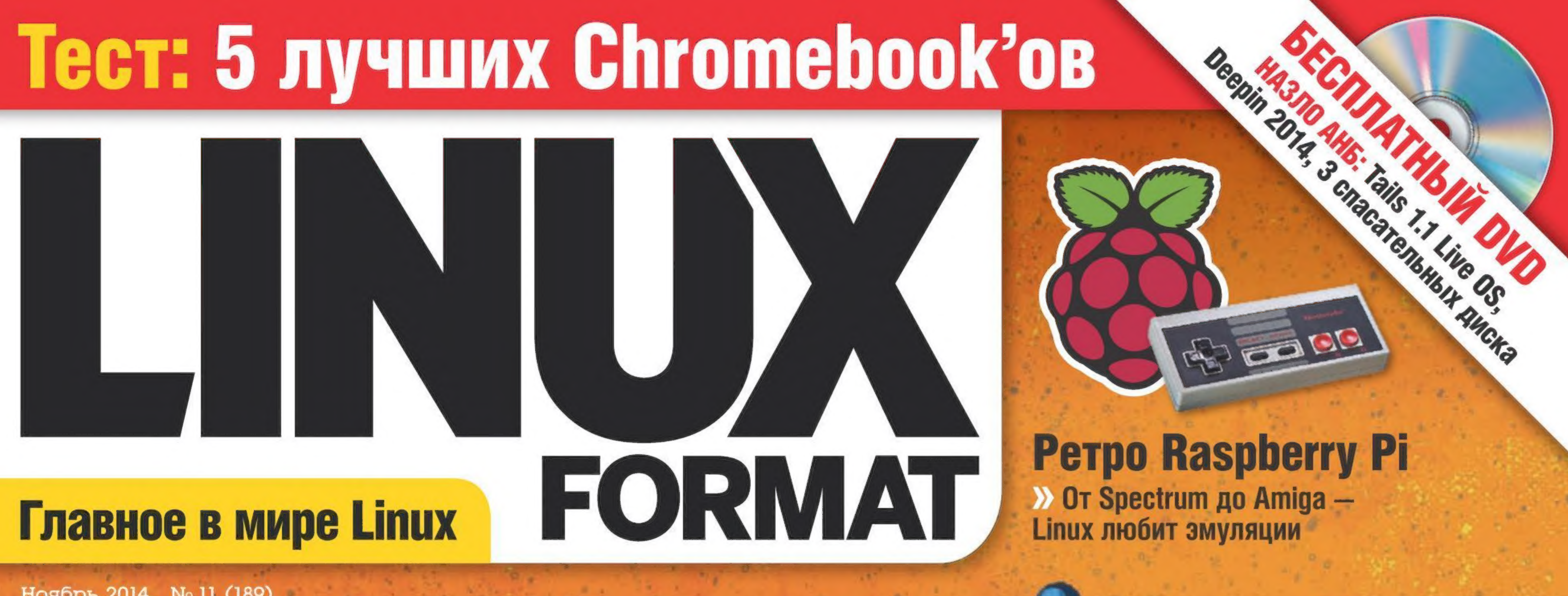
Тест: 5 лучших Chromebook'ов



Ретро Raspberry Pi

» От Spectrum до Amiga — Linux любит эмуляции

Ноябрь 2014 № 11 (189)

ВАШИ ПРОБЛЕМЫ В LINUX РЕШЕНЫ

» Рычаги Grub
» Вопросы драйверов
» Ошибки печати
» Сети
» Медленные ОС
» Журналы-логи

ЕСТЬ ЭЛЕКТРОННАЯ ВЕРСИЯ ДЛЯ ВАШЕГО ПЛАНШЕТА!

ПЛЮС! Прикрутим к ПК интегральные схемы с. 88

Высокие стандарты

«Следующая реализация будет уже не эталонной, но основной для всей страны»

Итало Виньоли Как OpenDoc завоевывает мир с. 52

Также в номере...

Linux на Android
» Любимая ОС угнездится на вашем Android-устройстве с. 28

Web-серверы
Динамический Nginx
» Украсьте свой web-сервер XHTML

VNC и SSH
Удаленный доступ
» Управляем своими ПК легко и безопасно

Код Python
Импорт в Minecraft
» Мигом увешайте стену картинами... и взорвите

ПОДПИСНЫЕ ИНДЕКСЫ В КАТАЛОГАХ
Агентство «Роспечать» — 36343,
«Почта России» — 11932, «Пресса России» — 90959







Что мы делаем



» Мы поддерживаем открытое сообщество, предоставляя источник информации и площадку для обмена мнениями.

» Мы помогаем всем читателям получить от Linux максимум пользы, публикуя статьи в разделе «Учебники»: здесь каждый найдет что-то по своему вкусу.

» Мы выпускаем весь код, появляющийся на страницах раздела «Учебники», по лицензии GNU GPLv3.

» Мы стремимся предоставлять точные, актуальные и непредвзятые сведения обо всем, что касается Linux и свободного ПО.

Кто мы

Этот номер посвящен решению проблем в Linux, и мы спросили наших авторов: какой была их худшая проблема?



Джонни Бидвелл
Моя забытая установка Ubuntu сломалась: ворчит, что раздел root смонтирован только на чтение, а затем выкатывает приветствие, что все отлично. А вот и нет! Я не мог сдвинуть курсор или что-то ввести… перезагрузился по Ctrl-Alt-SysRq-B, огорчив системный журнал. Да наплюйте, у меня и другие дистрибутивы есть.



Нейл Ботвик
Поскольку мои друзья и родные переходят на Linux, я превращаюсь в их техподдержку. Это вроде китайского обычая: спас кому-нибудь жизнь — потом за эту жизнь отвечаешь. Но худшие проблемы, похоже, вызваны пользователями, когда ОС вызывает одни проклятия, потому что не выдает реакции.



Дэвид Хейвард
С нежностью вспоминаю, как устанавливал на старом ПК с Red Hat прокси-сервер для колл-центра. Проблема была в том, что он работал только когда я присутствовал в здании. Стоило мне уйти, и через десять минут он падал. Причем каждый раз по совершенно непонятной причине.



Лес Паундер
Ничего не имею против *Samba*, но мы с ним видимся редко. Вплоть до нынешнего дня я использую поверх *Samba* NFS, SFTP и SSH. На мое счастье, у всех моих домашних установлен Linux — даже у миссис П., которая на своем ноутбуке держит Crunchbang Linux.



Маянк Шарма
Была у меня проблема в 1999 году: как попасть в графический рабочий стол на установке Slackware. Концепция того, что рабочий стол и ОС — это разные вещи, не укладывалась в моем 13-летнем мозгу. И я устанавливал и переустанавливал дистрибутив, удивляясь, почему постоянно попадаю в командную строку.



Валентин Синицын
KDE 4. Зачем, спрашивается, было чинить то, что не ломалось? Сейчас, по истечении шести лет, должен признать, что рабочий стол вышел неплохим, но в далеком 2008-м… KDE 4 появился в моем Arch Linux слишком быстро, а KDE Plasma 5 не появляется слишком долго.



Товарищ и man

» Вероятно, на каждом уважающем себя заводе есть своя легенда о старом мастере, обладателе некоего «тайного знания», без которого работа невозможна. И если мастер перед уходом на пенсию не передаст его подмастерьям, завод придется закрыть. Вот только в реальной жизни мне такие люди не встречались. Скорее — наоборот. Все известные мне настоящие мастера считали своим долгом помочь «младшим товарищам» дельным советом.

Сегодня такие советы можно найти на многочисленных тематических сайтах и форумах по любому вопросу — от разведения роз до газовой сварки. Но начиналось все пару десятилетий назад именно с «советов про ИТ». Традиция помогать друг другу и делиться опытом в нашей индустрии была всегда. Сеть стала для этого идеальным инструментом.

Вероятно, одна из причин такого отношения кроется в скорости обновления технологий. Оберегать «тайное знание» попросту бессмысленно: все равно через год устареет. Если сегодня я помогу кому-нибудь, завтра он поможет мне. Как гласит старая админская шутка, админ админу — друг, товарищ и **man**.

Кирилл Степанов
Главный редактор
» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Проблемы с дисками: disks@linuxformat.ru
Вопросы распространения: sales@linuxformat.ru
Сайт: www.linuxformat.ru, группа «ВКонтакте»: vk.com/linuxform
› **Адрес редакции:** Россия, Санкт-Петербург, пр. Медиков, 5, корп. 7
› **Телефон редакции:** (812) 309-0686. Дополнительная информация на с. 112

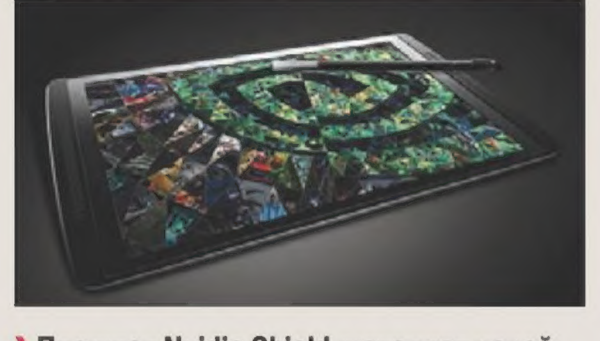


Содержание

«Проблему не решить на том уровне мышления, который ее породил».



Обзоры



› Планшет Nvidia Shield щеголяет новейшими мобильными технологиями.



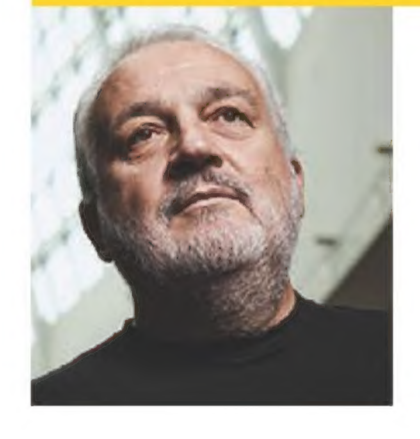
Ваши проблемы с Linux решены!

Кривые драйверы или старая добрая паника ядра? Срочно в наше 14-страничное руководство на с. 38!

Сравнение: Chromebook'и с. 32



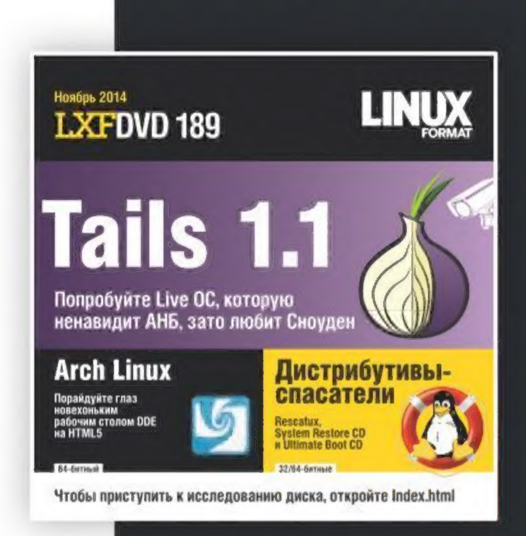
Люди говорят



> «Это уже история, а историю человечество способно изменять»

Итало Виньоли и стандарт Open Document с. 52

На вашем бесплатном DVD

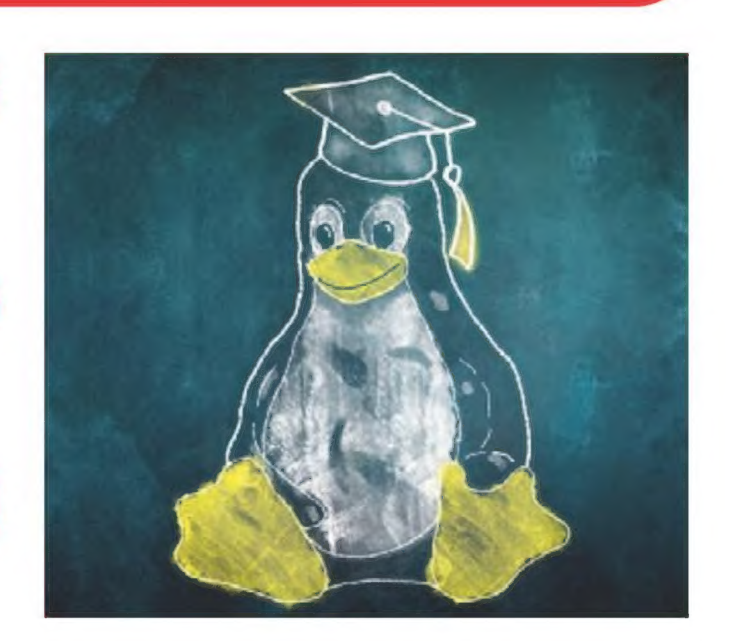


Tails 1.1
» Дистрибутив с упором на безопасность

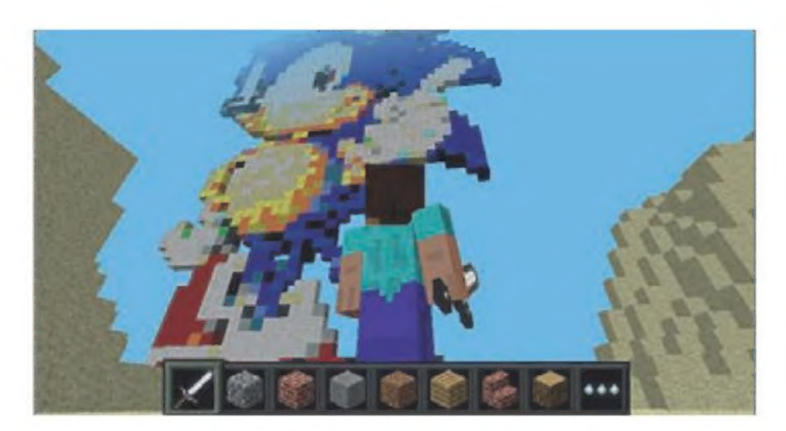
Deepin 2014
» Соперник из Китая — это серьезно

ПЛЮС: HotPicks и коды учебников с. 106

Ищите в этом номере



Академия кодинга



Постоянные рубрики

Учебники

› *Shotcut* безусловно стоит знакомства.



» Рубрику готовил **АНДРЕЙ ГОНДАРЕНКОВ**

МОБИЛЬНЫЕ СИСТЕМЫ

Объединяют усилия

«РОСА» и OpenMandriva нашли новые точки сотрудничества.

Уже не один год компания «РОСА» (ООО «НТЦ ИТ РОСА») плотно сотрудничает с некоммерческой ассоциацией OpenMandriva (OMA) в работе над настольными ОС. Новый виток взаимодействия связан с темой мобильности: теперь известная российская ИТ-компания и OMA совместно создают мобильную версию ОС для планшетов на базе ARM.

OpenMandriva Association уже некоторое время изучает возможности предоставления пользователям мобильного решения. Объединение наработок «РОСА» и OMA в совместном репозитории позволит достигать синергии в дальнейшей разработке. Специалисты «РОСЫ» в рамках этого сотрудничества продолжат дорабатывать элементы, связанные с безопасностью и обеспечивающие возможность сертификации системы на соответствие российским требованиям по защите информации.

В дальнейшем сотрудничество может быть расширено и на вариант мобильной системы для архитектуры x86, которую «РОСА» пока ведет своими силами на основе пакетной базы ROSA Desktop.

Совместная работа ведется с использованием общей системы разработки и сборки ROSA ABF, благодаря чему программистам легко обмениваться наработками между командами и интегрировать результаты. Кроме того, ряд сотрудников «РОСА» напрямую участвует в сообществе OpenMandriva в качестве контрибьюторов.

ВНЕДРЕНИЯ

Европа выбирает свободу

«Призрак бродит по Европе, призрак Open Source!»

Слегка перефразированное выражение классика как нельзя лучше характеризует ситуацию: один за другим, регионы или министерства европейских стран переходят на СПО в своём документообороте. Список «перебежчиков» растёт не по дням, а по часам.

В Италии администрация г. Турина решила заменить на 8300 своих ПК устаревшую проприетарную ОС открытой альтернативой — Ubuntu, что за ближайшие пять лет сэкономит €6 млн. В г. Удине *Apache OpenOffice* уже стоит на всех 900 городских ПК, а в 2015-м стартует пилотный проект по использованию только открытого ПО. В госучреждениях г. Тоди переводят ПК на *LibreOffice*; за ним последует г. Терни. При участии LibreUmbria, организации, содействующей госадминистрациям по внедрению СПО, на *LibreOffice* перешли администрация Перуджи (1200 ПК) и региональный Минздрав (600 ПК). Администрация Генуи использует СПО везде, где возможно. Реализуется пилотный проект, который покажет, сможет ли GNU/Linux заменить проприетарную ОС на городских ПК. На Linux уже работают городские серверы и ряд публичных точек доступа в Интернет. 63 компьютера с Debian GNU/Linux стоят в местных школах, и городские ПК переводят на *LibreOffice*.

В Министерстве юстиции Австрии *Apache OpenOffice* с 2008 г. стал стандартом, ныне на 12 000 ПК. *OpenOffice* прекрасно адаптирован к потребностям Центра обработки данных Министерства, интегрирован с его спецприложениями и допускает автоматическое и полуавтоматическое создание и представление документов. Одна из услуг Центра — *Judok*, web-сервис создания документов.

«Гамбург, как Мюнхен, должен перейти на СПО! — считают в местной организации „Союз '90/Зелёные". — СПО дает дорогу инновациям и повышает безопасность, что должна учесть городская администрация при выборе решений по ICT». 21 октября при содействии альянса Open Source Business (OSB) в городе прошла конференция по СПО для бизнеса, общества и госуправления — параллельно саммиту ИТ от Министерства экономики и энергетики с участием канцлера Ангелы Меркель [Angela Merkel]. По мнению Зелёных, Германии нужен "Open IT Summit", раз СПО пока не принято в стране: «Опора на монополистов ИТ-систем обойдётся нам дороже».



» Италия возводит *OpenOffice* на свой государственный флаг.

ЯЗЫК КОДА ЯДРА

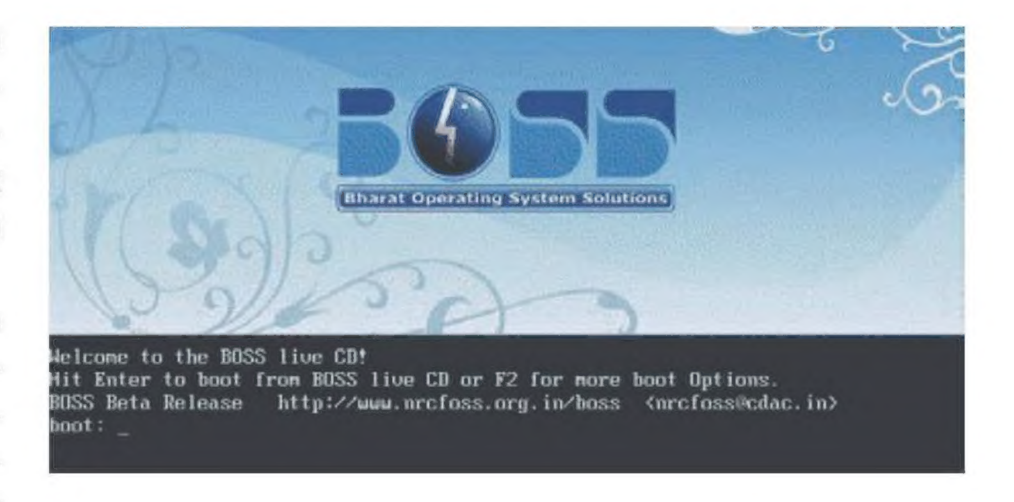
BOSSMOOL

Предлагается объектно-ориентированная парадигма для Linux.

Основной причиной выбора процедурного языка С для написания кода ядра Linux была его эффективность. Но данный выбор привёл к высокой степени зависимостей (связей) между различными частями ядра Linux, затруднив поддержку и сопровождение кода. Индийские разработчики решили, что объектно-ориентированный (ОО) подход позволит реализовать всё намного проще.

CDAC (Centre for Development of Advanced Computing), основная компания в IT-индустрии Индии, уже много лет развивает собственный дистрибутив GNU/Linux на базе Debian — BOSS (Bharat Operating System Solution), адаптируя его для индийских условий (в частности, поддерживая почти все местные языки). Сотрудники лаборатории DOS (Distributed and Object Systems) Индийского технологического института в Мадрасе [Indian Institute of Technology Madras] создали свою версию ядра Linux, MOOL (Minimalistic Object Oriented Linux, «минималистичный ОО-Linux»). Ныне сотрудничество этих двух команд направлено на создание нового дистрибутива — BOSSMOOL.

В ядре MOOL будут функции, знакомые по ОО-парадигме. Сокращается число глобальных переменных. Типовые такие переменные, используемые несколькими модулями, заменят на передачу значений как аргументы функции. MOOL также располагает фреймворком для написания драйверов устройств на языке C++ в виде загружаемых модулей ядра. Кроме того, ядро BOSSMOOL поддерживает локализацию на уровне консоли. Возможно создание реализуемых в виде модулей ядра фильтров Message Filter — ОО-обвязок для перехвата системных вызовов, обеспечивающих расширяемость без изменения существующего кода ядра. Любой системный вызов пользователя проходит через такой фильтр и может быть перехвачен им. Фильтры могут осуществлять мониторинг в реальном времени, применять политики безопасности, обрабатывать отказы драйверов устройств и т.д.



› Этот Linux с индийским лицом стал родоначальником ОО-дистрибутивов.

Создатели BOSSMOOL считают свой проект крупным прогрессом в разработке СПО. Правда, Торвальдс C++ не одобряет.

FSF И DEBIAN

Совместимое «железо»

Для пользователей СПО упростится поиск требуемого оборудования.

Фонд Free Software Foundation (FSF) и Проект Debian сообщили о совместной работе по расширению и улучшению h-node — базы данных, помогающей пользователям узнавать и делиться информацией о компьютерах, которые работают со свободными ОС.

Другие базы данных содержат списки оборудования, совместимого с GNU/Linux; h-node же считает оборудование совместимым, только если для его работы не требуются проприетарное ПО или микропрограммы. Информация об оборудовании, не прошедшем этой проверки, тоже включена в базу, чтобы пользователи знали, чего избегать. Указываются и отдельные компоненты, типа карт Wi-Fi и видеокарт, и целые системы, например, ноутбуки.

Сведения о совместимости поступают от пользователей, тестирующих оборудование на системах с чисто свободным ПО. Ранее правила сайта h-node требовали от пользователей применять один из одобренных FSF дистрибутивов. FSF не включает в этот список Debian (поскольку проект Debian предоставляет также репозиторий с несвободным ПО), но подтверждает, что основной репозиторий Debian — по умолчанию единственный источник пакетов — полностью свободен.

«В отличие от других дистрибутивов GNU/Linux, установка официального выпуска Debian по умолчанию подразумевает установку только свободного ПО. Пока пользователи Debian не добавляют репозитории с добавочными пакетами, их системы являются надёжным источником информации о полной совместимости с СПО. Мы рады сотрудничать с Debian ради помощи пользователям СПО в получении подходящего им оборудования и стимулирования компаний к поддержке СПО», говорит исполнительный директор FSF Джон Салливан [John Sullivan]. «Сотрудничая с h-node, Debian получил возможность присоединиться к попыткам других сообществ СПО по созданию базы данных оборудования, для корректной работы которого не требуется что-либо за пределами основного архива Debian, — заявил лидер Проекта Debian Лукас Нуссбаум [Lucas Nussbaum]. — Мы приглашаем всех членов сообщества Debian принять участие в этом проекте и выслать информацию о своём оборудовании на сайт h-node».

› Свободные программы и оборудование — свободное общество (Проект GNU).



Проект h-node основал Антонио Галло [Antonio Gallo], ныне главный его разработчик. Фонд FSF предоставляет инфраструктуру и поддержку. Сайт работает на СПО под лицензией GNU GPL v3.

Информация об участии в проекте пользователей и разработчиков может быть найдена по адресу http://h-node.org/help/page/en/Help.

СДЕЛКИ

Правь, Британия?

Компании Attachmate Group и Micro Focus объявили о слиянии.

В $1,2 млрд оценивается сделка по поглощению холдинга Attachmate Group (в чей состав входят в т. ч. Novell и SUSE) британской компанией ИТ-сектора Micro Focus International.

Attachmate Group — частная холдинговая компания-производитель ПО, принадлежащая инвестиционной группе во главе с Francisco Partners, Golden Gate Capital, Elliott Management и Thoma Bravo. Ожидается, что документально все операции по слиянию будут завершены к 3 ноября 2014 года. Сделка пройдет по стандартным условиям закрытия, а также с согласия акционеров Micro Focus и регулирующих органов в соответствии с Актом Hart–Scott–Rodino.

Attachmate Group и Micro Focus — давно и хорошо зарекомендовавшие себя ИТ-компании, с узнаваемыми послужными списками решений; они помогли многим клиентам лучшим образом организовать свою ИТ-инфраструктуру. Объединенная новая компания, с ежегодным доходом в $1,4 млрд, будет представлена на всех крупнейших мировых рынках и продолжит разрабатывать передовые решения во многих областях ИТ.

«В итоге слияния Attachmate Group и Micro Focus значительно расширится спектр предлагаемых решений, расширится глобальное присутствие и укрепится финансовая устойчивость, — отметил Джефф Хоун [Jeff Hawn], председатель совета директоров и главный исполнительный директор Attachmate Group. — Совместная компания будет эффективнее помогать своим заказчикам в решении их текущих бизнес-задач при помощи ИТ, с учетом усложняющихся условий ведения бизнеса, текущих изменений и рисков».

Бренды компании Attachmate Group — Attachmate, NetIQ, Novell и SUSE — предлагают огромный выбор взаимодополняющих продуктов по решению таких задач:

- управление ИТ-процессами;
- решения Linux и Open Source;
- сервисы файлов и сети, совместная работа и управление конечными точками;
- подключение хостов и модернизация имеющейся ИТ-инфраструктуры;
- идентификация и управление доступом, решения по безопасности;
- управление рабочими нагрузками и облачные вычисления; и другие.

«Начинается интереснейшее время для двух организаций, для их заказчиков, акционеров и сотрудников, — сказал Кевин Лузмор [Kevin Loosemore], исполнительный директор и председатель совета директоров Micro Focus. — Мы глубоко уважаем компанию Attachmate Group и ее портфель брендов. Масштаб и опыт наших компаний будет способствовать миссии, нацеленной на помощь клиентам по сохранению бизнес-логики и данных заказчиков, используя при этом облачные, мобильные и другие передовые ИТ-решения на глобальном уровне».

ОФИСНЫЕ ПАКЕТЫ

Слияние как решение проблем

Open и *Libre Office* должны объединиться?

Глава ИТ-отдела Федерального верховного суда Швейцарии подвёл итоги 10 лет использования в этой государственной структуре СПО.

Разработчикам *Apache OpenOffice* и *LibreOffice* — двух тесно связанных свободных офисных пакетов — следует преодолеть существующий раскол и объединиться, создав полноценного конкурента повсеместно распространённой проприетарной альтернативе, объявил Даниэль Бруннер [Daniel Brunner], глава ИТ-отдела Федерального верховного суда Швейцарии, выступая на Конференции *LibreOffice* в Берне. Свою презентацию, посвящённую 10-летию использования открытых офисных приложений в Верховном суде Швейцарии, г-н Бруннер был вынужден предварительно протестировать в обоих свободных Офисах, т. к. нынешнее состояние их разработки ведёт ко всё большей несовместимости. По его мнению, слияние этих двух проектов станет для многих госучреждений прекрасным побудительным мотивом к переходу на открытый офисный пакет.

Федеральный верховный суд Швейцарии сейчас использует *OpenOffice*, который, уверен г-н Бруннер, лишь выиграл бы от добавления улучшенных фильтров документов, реализованных в *LibreOffice*. Преимущества *OpenOffice* — бóльшая стабильность и доступность для мобильных платформ, но *LibreOffice* располагает более многочисленным сообществом разработчиков и предлагает новейшую функциональность.

Федеральный суд Швейцарии — один из ярких примеров применения СПО органами госуправления: все 460 настольных ПК этой структуры работают с *OpenOffice*. Среди положений реализуемой здесь стратегии использования открытых IT-стандартов — требование независимости программ от поставщиков: «Мы хотим иметь возможность переключаться между



› Не пора ли Офисам прекратить «разброд и шатание»? Выиграет пользователь!

дистрибутивами GNU/Linux». Применяемые открытые решения должны быть распространёнными: «Мы не хотим привязываться к узкоспециализированному приложению Open Source».

IT-структура Федерального суда создана на базе сервера Red Hat Linux и комплекса для автоматизации коллективной работы *Zimbra*. Для удалённого доступа служит *Thinlinc*, основанный на *TigerVNC*. Настольный инструментарий включает email-клиент *Thunderbird* и браузер *Firefox*.



РЕЛИЗЫ

Red Hat Satellite 6

Упрощается развертывание и управление RHEL.

Вышел очередной стабильный релиз Red Hat Satellite 6, платформы для системного управления Linux-инфраструктурой, упрощающего развертывание, масштабирование, совместимость и менеджмент системы Red Hat Enterprise Linux. Это комплексное решение обеспечивает полное управление жизненным циклом и контентом систем Red Hat, патчами и конфигурациями, а также подписками для физических, виртуальных и облачных окружений при улучшенной доставке системы. Satellite предназначен не только для RHEL, но и для таких продуктов, как RHEL OpenStack Platform, Red Hat Enterprise Virtualization, Red Hat Storage, Red Hat Directory Server, Red Hat Certificate System и OpenShift Enterprise.

Среди новых возможностей, реализованных в Red Hat Satellite 6:

» обнаружение, предоставление и апгрейд всей инфраструктуры «голого железа»;

» создание и контроль сущностей в публичных и частных облаках;

» интеграция инструмента *Puppet* для поддержки управления конфигурациями;

» улучшенные методы управления контентом за счёт применения модулей Red Hat Package Managers (RPMs) и *Puppet*;

» объединение средств управления средой RHEL, от предоставления и конфигурирования до распространением и доставки контента

В отличие от версии 5 и более ранних, на базе свободного проекта Spacewalk, Satellite 6 уходит корнями к открытым решениям Foreman, Pulp, *Puppet*. Для миграции на Satellite 6 с предыдущих версий компания предоставляет пользователям специальный инструментарий. Джо Фицджеральд [Joe Fitzgerald], генеральный менеджер Red Hat по облачному управлению, обращает внимание на расширенные возможности Satellite 6 по управлению облачными системами, включая инициализацию и слежение, будь то серверы bare metal или же виртуальные, публичные, гибридные либо частные облачные структуры.

ДИСТРИБУТИВ

ROSA Fresh KDE R4

Отечественный настольный дистрибутив обновился.

«РОСА» представила дистрибутив ROSA Desktop Fresh с рабочим столом KDE. В репозитории выложена большая коллекция игр и эмуляторов, можно установить *Steam*, и привычное ПО для текстовой и видеокоммуникации, включая *Skype*. Работает просмотр видео современных форматов. В дистрибутив входит свежий *LibreOffice 4.3.1*, но в репозиториях есть полный стек TeX и все известные Open-Source программы для документирования, верстки и полиграфии. Разработчики получат полноценные окружения LAMP/C++/... для разработки.

Дистрибутив может быть использован как домашними пользователями для повседневной коммуникации и развлечений, так и пользователями с особыми требованиями, способными самостоятельно позаботиться о настройке и постоянной работоспособности системы. Дистрибутив основан на новой платформе rosa2014.1 со сроком поддержки 2 года; на базе этой платформы будет выпущено несколько релизов дистрибутива, аналогично платформе rosa2012.1.

Основной рабочий стол, KDE, обновлен до версии 4.13.3, а популярные пользовательские приложения — до свежих версий. В дистрибутив добавлена экспериментальная версия ROSA Freeze — инструментария для надежного отката изменений системы, произведенных пользователем во время работы.

Ядро дистрибутива, LTS-3.14.x, собрано с поддержкой BFQ-планировщика версии 7r5. Доступны 3 стабильных версии проприетарного драйвера видеокарт Nvidia: 340.x, 304.x и 173.x, а инструмент настройки графической подсистемы, *XFdrake*, сам выбирает подходящий драйвер. Улучшена поддержка систем с гибридной графикой Nvidia+Intel. LXF

Новости короткой строкой

» Вышла первая за 7 лет стабильная версия набора приложений для высокоуровневого интерфейса к разным пакетным менеджерам — *PackageKit 1.0.0*. Источник: permalink.gmane.org

» Библиотеки Enlightenment Foundation Libraries (EFL) обновлены до версии 1.11, а в новой версии рабочего стола *E19* улучшена поддержка *Wayland*. Источник: https://phab.enlightenment.org

» В *digiKam 4.4.0* улучшено управление слайд-шоу; эскиз альбома стал отображаться во всплывающей подсказке. Источник: https://www.digikam.org

» Настольная среда Gnome 3.14 полноценно заработала с *Wayland*, а также с мультитач и протоколом DLNA. Источник: www.gnome.org

» Вышел дистрибутив сообщества OpenMandriva Lx 2014.1 — третий под эгидой OpenMandriva Association. Источник: https://blog.openmandriva.org

» Adobe представила *Photoshop Streaming* — запускаемое в браузере и выполняемое в виртуализованном окружении Adobe Cloud web-приложение Chrome OS. Источник: chrome.blogspot.ru

» К 15-летию свободной системы печати CUPS выпущен релиз 2.0 с поддержкой *systemd*, проверкой TLS-сертификатов и повторной отправой задания на печать в виде растрового файла при неудаче с отправкой файла высокоуровневого формата. Источник: www.cups.org

» Новая версия ядра, Linux 3.17, содержит 12354 исправления. 46% изменений — драйвера устройств, 18% — код аппаратных архитектур, 12% — сетевой стек, 5% — файловые системы, 3% — подсистемы ядра. Источник: https://lkml.org

» Ubuntu 14.10 (Utopic Unicorn) обновил ПО, улучшил производительность и снизил системные требования. Источник: www.ubuntu.com

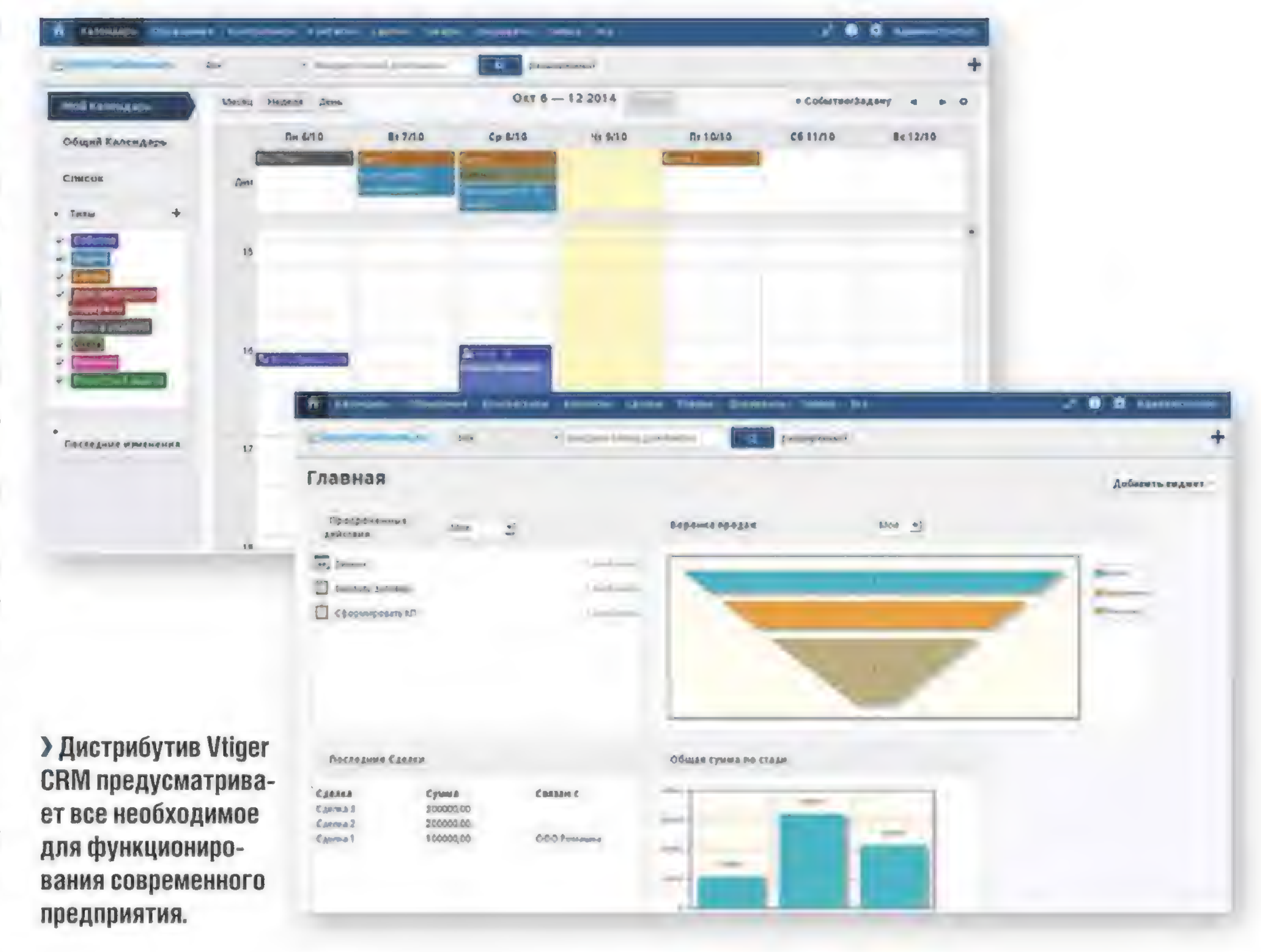
» Организация Open Knowledge и консультативный совет проекта Open Definition определили критерии открытой информации в Open Definition 2.0. Источник: https://creativecommons.org

» CyanogenMod 11.0 M11 ввел поддержку ряда новых устройств Linux. Источник: www.cyanogenmod.org

Дистрибутив для предприятий

Российский разработчик SalesPlatform выпускает новую версию свободной CRM.

Дистрибутив SalesPlatform Vtiger CRM 6.1, уже доступный для загрузки, построен на основе свободной платформы Vtiger CRM, хорошо зарекомендовавшей себя на мировом рынке. Система позволяет автоматизировать весь цикл операционной деятельности предприятия: процессы продаж и маркетинга, включая телефонные звонки клиентам, email- и SMS-рассылки, регистрацию заявок с корпоративного сайта, формирование документов (коммерческие предложения, счета, акты, счета-фактуры, накладные), ведение номенклатуры товаров и услуг, закупки, а также процессы обработки заказов клиентов и послепродажного обслуживания. В дистрибутиве SalesPlatform все эти функции поддерживают русскоязычный интерфейс и адаптированы к особенностям ведения бизнеса в РФ.



› Дистрибутив Vtiger CRM предусматривает все необходимое для функционирования современного предприятия.

Как отмечают разработчики компании SalesPlatform, версия 6.1 обладает отлаженным пользовательским интерфейсом на основе современных web-технологий. Благодаря web-интерфейсу система может быть использована сотрудниками предприятия на любых устройствах — от персональных компьютеров и ноутбуков до интернет-планшетов и смартфонов. Также не представляет труда развертывание системы на предприятиях с несколькими филиалами или во франчайзинговых сетях. Для подключения к системе удаленного филиала достаточно стандартного доступа в сеть Интернет, при этом гибкая система прав доступа позволит разделить информационные базы филиалов, а также управлять доступом для различных ролей сотрудников, вплоть до отдельных полей данных.

За счет интеграции со свободной IP-АТС *Asterisk*, SalesPlatform Vtiger CRM приобретает функции управления исходящими и входящими звонками сотрудников или операторов колл-центра, включая быстрый набор номера по информации из клиентской базы, отображение карточки клиента при входящем вызове, хранение истории звонков с привязкой к клиентам, отчеты, возможность прослушивания записанных разговоров непосредственно из интерфейса CRM.

Компании, которые используют для ведения бухгалтерского учета систему «1С: Предприятие», могут осуществить интеграцию SalesPlatform Vtiger CRM с учетной системой как на основе протокола CommerceML, так и с использованием web-сервисов SOAP. Такая интеграция позволяет разделить деятельность сотрудников, отвечающих за продажи и работу с клиентами, и внутренний учет предприятия, обеспечив безопасный и надежный механизм взаимодействия систем без предоставления внешнего доступа к системе внутреннего учета из сети Интернет. Протокол CommerceML может быть использован и для интеграции SalesPlatform Vtiger CRM с корпоративным сайтом или интернет-магазином.

Дополнительные удобства могут получить пользователи за счет интеграции CRM с сервисами Google. Объединив сервис календарей Google и встроенный календарь событий и задач SalesPlatform Vtiger CRM, пользователи получат мощный и удобный инструмент для тайм-менеджмента и совместной работы, который удобно использовать как на офисном компьютере, так и на персональных мобильных устройствах сотрудников.

Специально для российского рынка разработчики SalesPlatform значительно улучшили поддержку русского языка, а также реализовали модули для ведения первичной бухгалтерской документации по стандартам РФ, модули интеграции с системой «1С: Предприятие» и социальными сетями.

Российским заказчикам компания SalesPlatform предоставляет услуги по технической поддержке и сопровождению, а также услуги по внедрению и разработке индивидуальных решений, включая реализацию проектов по интеграции CRM с другими информационными системами предприятия, такими как системы электронного документооборота, корпоративные сайты, биллинговые системы и др.

Дистрибутив SalesPlatform Vtiger CRM можно бесплатно загрузить с сайта salesplatform.ru. Также разработчики предоставляют русскоязычную документацию по системе и ресурсы для сообщества пользователей и специалистов по внедрению системы. LXF

LINUX FORMAT

Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов



АЛЕКСЕЙ ФЕДОРЧУК
Тэг <сарказм> по умолчанию, смайлики по вкусу.

Дистрибутив для себя

Перефразируя Черчилля, можно сказать, что линуксоид, не мечтавший в юности о собственном дистрибутиве, не имеет сердца. Продолжающий же мечтать о нём в зрелости — не имеет разума. Ибо все разумные идеи дистростроения давно реализованы. Впрочем, большинство неразумных — тоже. Другое дело — кастомизация существующих дистрибутивов с точки зрения комплектации приложениями: она во многом — дело вкуса, который не обязан совпадать у майнтайнеров и применителей.

В принципе, в кастомизации готового образа нет ничего сверхъестественного. Но это кропотливая работа, требующая аккуратности. И потому очень важно наличие инструментов, способных её упростить. Одним из таких инструментов является *Open Build System*, позволяющий собирать собственные ремиксы на базе широкого круга дистрибутивов. А для семейства Ubuntu и её клонов есть более простое средство — *Ubuntu Customization Kit* (*UCK*). С его помощью индивидуализированный ремикс на базе Mint'а, в том числе и с «нескучными обоями», можно сделать за полчаса — если, конечно, знать, что делать. Информацию по работе с *UCK* в Mint'е можно найти здесь: http://alv.me/.

Применение *UCK* не ограничивается удовлетворением личных амбиций. С его помощью можно собирать ремиксы под любые специальные задачи — от ремонтно-восстановительных до ориентированных на тяжёлые приложения типа *GIS*. Причём делать это не просто, а очень просто.

alv@posix.ru

Сегодня мы рассматриваем:

› Nvidia Shield: сверкает мощным мобильным процессором Tegra K1 от Nvidia.

UberStudent 4.0

› Учись, студент! Для этого предоставляются все возможности.

KDE 5.0



› Отличная новость: очередной главный релиз KDE ничего не поломал.

Сравнение: Chromebook'и с. 32

Acer C720P



Asus C200



HP 14-q005sa



Samsung XE503C12



Toshiba CB30-102



С момента их появления в 2011 году Chromebook'и принялись плодиться как кролики. Мы протестировали пять лучших, которые извлекают все преимущества ChromeOS от Google.

UberStudent 4.0

Маянк Шарма выясняет, сможет ли этот ориентированный на образование дистрибутив сделать новичков отличниками.

Вкратце

» **Дистрибутив на базе *Xfce*, набитый образовательными приложениями и инструментами. См. также: Edubuntu и Skolelinux.**

Создателем UberStudent руководит профессиональный педагог; видимо, именно поэтому дистрибутив претендует на большее, чем просто стать очередной коллекцией образовательных программ. Разработчики постарались сделать его доступным и открытым для всей академической сферы; на самом деле, одна из целей его создания — содействие компьютерной грамотности.

Свежая версия, UberStudent 4 — на основе Ubuntu релиза 14.04 LTS и слегка изменённого рабочего стола *Xfce*. Все образовательные приложения включены в пункт меню Образование, с подкатегориями типа Исследование, Письмо, Учебные пособия. Для программ, не имеющих документации, разработчики прочесали Интернет, добавив ресурсы в подменю Документация. Выполнена преднастройка сложных приложений, вроде исследовательского инструмента *Zotero*, чтобы с ними можно было работать «из коробки».

Помимо этого, есть ряд интернет-приложений. Например, меню Книги ссылается на приложение *FBReader* для чтения электронных книг, а также ряд ресурсов, откуда можно эти книги скачать, типа Open Library. Кроме того, категория Личные финансы содержит ссылку на онлайн-приложение *Mint Personal Financial Manager*, которое помогает уменьшить расходы студента, указывая сайты, где можно купить книги б/у или одолжить новые.

Дистрибутив включает и кучу приложений общего назначения, включая инструменты воспроизведения и создания мультимедиа, а также средства передачи файлов, такие как *Dropbox*, и даже клиент *Steam*. Если этого мало, можно добыть ещё — в меню почти каждого приложения есть своя ссылка на установщик.



› Тема значков UberOxygen — ответвление KDE Oxygen, с десятками новых.

Приложения берутся из собственных репозиториев дистрибутива — помимо репозиториев родителя-Ubuntu, это также официальные репозитории Dropbox, Google, Oracle и Videolan. В дистрибутиве есть свой менеджер обновления и источники приложений, основанные на Mint, а не на Ubuntu. Управление пакетами реализуется средствами *Synaptic* и *Ubuntu Software Center*. Браузеры *Firefox* и *Chromium* оснащены плагинами Java и Flash; также можно установить дополнительные кодеки по ссылке в окне приветствия.

Для всех и каждого

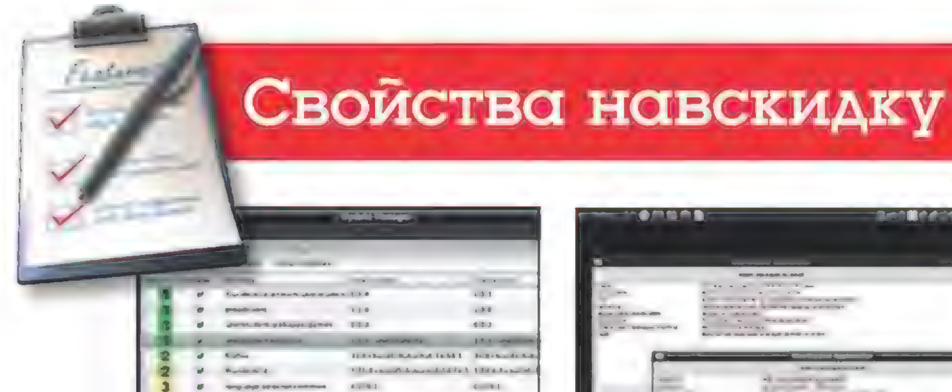
Все это реализует инициативу ориентировать дистрибутив на новичков. Здесь есть и скрипт восстановления системы, работающий в фоновом режиме во время загрузки и решающий распространённые проблемы по получаемым запросам на поддержку. Стандартный рабочий стол *Xfce* выполнен в традиционном стиле, и в его составе — диспетчер файлов *Thunar* с привычной функциональностью. Есть предпросмотр мультимедиа-файлов без их открытия с помощью *Gloobus Preview*, а затем их можно одним щелчком переправить в указанное место; например, в папку *Dropbox*.

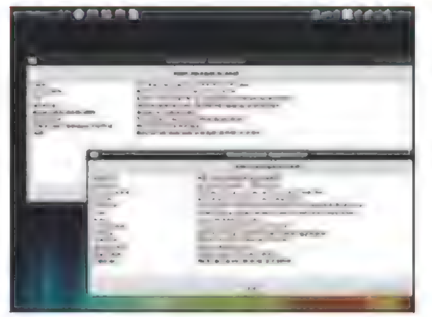
Дистрибутив также использует обои с встроенным скриптом *Conky*, чтобы следить за состоянием системных ресурсов. Для красоты, вместо родного инструмента композитинга *Xfce* компоновку изображений осуществляет более лёгкий *Compton*.

Давненько нам не встречался дистрибутив, который не дрогнул бы под натиском нашего тестирования. Создание хорошо интегрированного и логичного простого дистрибутива — задача сама по себе не простая. Но UberStudent удалось добиться даже большего: они сгладили шероховатости настольного Linux и сделали его более презентабельным.

Дистрибутив четко нацелен на студентов и преподавателей, но явно подойдёт не только им. Здесь есть все приложения для повседневной работы, в т.ч. несколько проприетарных. UberStudent 4.0 — хорошо продуманный вариант Ubuntu на базе *Xfce*, достойный конкурент популярным настольным дистрибутивам. LXF

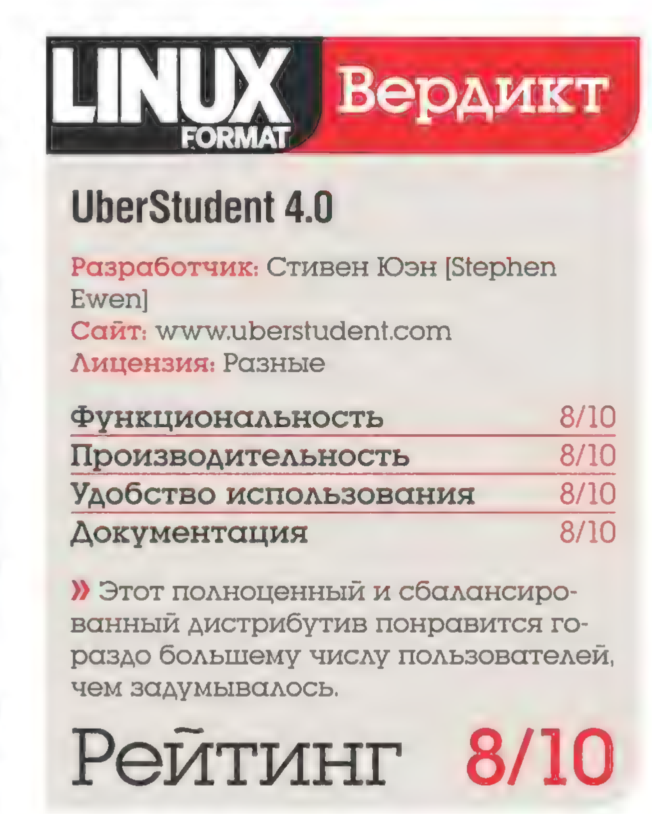
Свойства навскидку





Сверхнастройка
Разработчики настроили каждый аспект дистрибутива, чтобы он стал более доступным и дружелюбным к пользователю.

Универсальность
Помимо образовательных инструментов, в UberStudent есть всё, что полагается обычному дистрибутиву.



LINUX FORMAT Вердикт

UberStudent 4.0

Разработчик: Стивен Юэн [Stephen Ewen]
Сайт: www.uberstudent.com
Лицензия: Разные

Функциональность	8/10
Производительность	8/10
Удобство использования	8/10
Документация	8/10

» Этот полноценный и сбалансированный дистрибутив понравится гораздо большему числу пользователей, чем задумывалось.

Рейтинг 8/10

KDE 5.0

Маянк Шарма обжег пальчики на KDE 4.0 в 2008 году и теперь берется за новый «круглый» релиз не без трепета.

Вкратце

» **Один из самых популярных и богатых функциями рабочих столов Linux. См. также: Gnome и Unity.**

Предыдущий «круглый» релиз KDE взорвал пользователей Linux, которые прочили ему гибель и убегали на другие рабочие столы. Но KDE справился с неприятной реакцией, хорошея с каждой новой версией. Через шесть лет после релиза 4.0 вышла версия 5.0, и пользователи могут облегченно вздохнуть: рабочий стол Plasma в KDE 5.0 сохранил удобство последнего релиза и приобрел достаточно отличий от ветви 4.x.

Менеджер входа в систему и экран блокировки выглядят совершенно по-новому. Оба компонента связаны со свежей темой Breeze. Новый анимированный экран заставки симпатичнее прежнего. Заметнейшее из изменений рабочего стола — обновленная утилита запуска *Kickoff*. Она тоже оформлена в стиле новой темы и включает элементы других утилит запуска на нескольких вкладках. На вкладке Favourites [Избранное] находится список часто используемых приложений. Щелкнув на любом приложении на вкладке Applications [Приложения] правой кнопкой мыши, можно прикрепить его к вкладке Favourites. На других вкладках находятся последние использованные файлы, популярные каталоги и съемные устройства. В новом меню *Kickoff* больше не присутствует поле для поиска. Желая найти файл или приложение, просто начинайте набирать. Окно с результатами поиска откроется автоматически.

Разработчики также перетасовали компоненты на рабочем столе. В левом верхнем углу есть меню, с помощью которого



› **KDE 5.0 закладывает почву для сближения интерфейсов пользователя на различных устройствах, автоматически подстраиваясь под устройство.**

можно переключать задачи [Activities] и добавлять виджеты. В правом нижнем углу вместо кэшью появилась новая иконка, но с похожими функциями — через нее добавляются виджеты на панель. Область оповещений также подверглась очистке. В ней стало меньше всплывающих окон и улучшились переходы между различными элементами.

Эволюционность

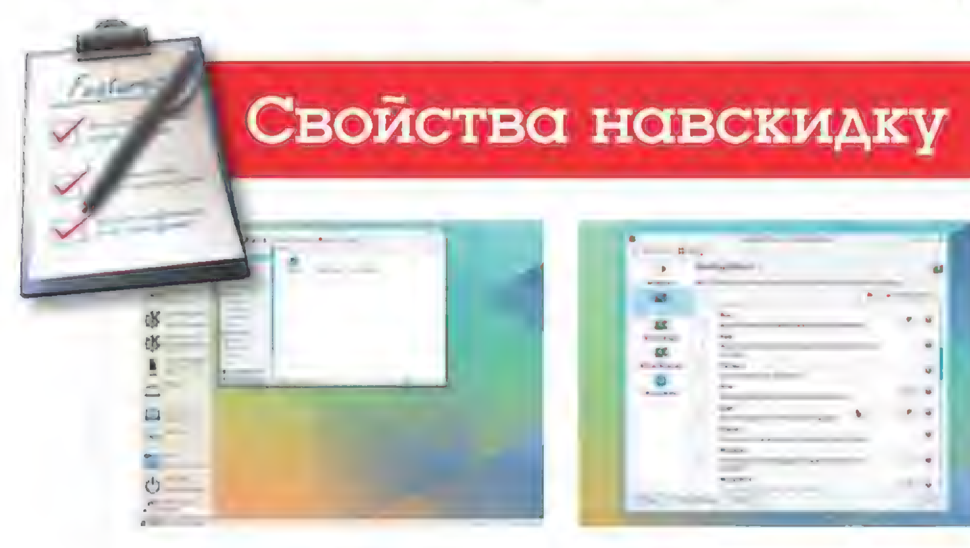
В KDE 5.0 также появилась новая схема переключения окон по умолчанию. При нажатии Alt+Tab для переключения между открытыми окнами на экране слева появляется панель со списком всех открытых окон для данного виртуального рабочего стола. Это лишь одно из возможных действий по переключению задач. Вместо боковой панели можно переключать окна перетаскиванием или горизонтальным вращением. Все дополнительные элементы, такие как виджеты и задачи, находятся на левой панели вместо их традиционного положения внизу.

Окно настроек системы почти не изменилось, но прекрасно вписано в рабочий стол. Некоторые компоненты, например, Задачи и поведение окон [Window Actions and Behavior] и переключатель задач [Task Switcher], были переработаны и теперь выглядят гармоничнее. Есть и функциональные изменения. Например, при переключении в режим просмотра папок [Folder View] KDE отлично интегрирует содержимое папок в рабочий стол.

Одно из главных скрытых изменений касается улучшения графической производительности, ставшего возможным благодаря тому, что в KDE 5.0 графические интерфейсы реализованы через OpenGL и интенсивные задачи по выводу графики можно передавать графическому сопроцессору. Это не только разгружает центральный процессор, но и позволяет получить более высокую частоту кадров и плавную графику. Также произошли изменения в поддержке KDE протокола графического сервера Wayland, который будет реализован в грядущем релизе.

Хотя KDE 5.0 скорее предназначен для знакомства с новыми изменениями, он очень удобен, отточен и является огромным шагом вперед по сравнению с предыдущим релизом. Он быстр, элегантен и содержит все, что должно быть в «круглом» релизе. LXF

Свойства навскидку



Новый свежий рабочий стол
Рабочий стол предложил новую свежую тему и реформировал изрядное количество важных компонентов.

Лучшая производительность
Рабочий стол перешел на графический стек с аппаратным ускорением, значительно улучшив производительность.



LINUX FORMAT Вердикт

KDE 5.0

Разработчик: KDE e.V.
Сайт: www.kde.org
Лицензия: GPL

Функциональность	9/10
Производительность	9/10
Удобство использования	8/10
Документация	8/10

» Быстрое, надежное и логичное обновление рабочего стола KDE Plasma.

Рейтинг 9/10

LXQt

Что получится, если скрестить Razor-qt и LXDE-Qt? **Джонни Бидвелл** исследует слияние двух рабочих столов-легковесов.

Вкратце

» **Легкий и тем не менее полнофункциональный рабочий стол на базе *Qt*, возникший при слиянии LXDEQt и Razor-Qt.**

В прошлом году команды разработчиков LXDE-Qt и Razor-qt объявили о планах объединить силы и «отобрать лучшее из Razor и LXDE». Первый официальный релиз новой системы, получившей название LXQt 0.7, вышел в мае и засвидетельствовал свой потенциал. Кроме того, команда LXDE сообщила, что в конце концов откажется от версии *GTK2+*, поскольку она чересчур устарела; однако *GTK3+* по сравнению с ней довольно «ожиревшая», что ставит под вопрос будущее дистрибутива Lubuntu на базе LXDE — впрочем, время покажет.

Как и в случае с Gnome 3.12, в большинстве дистрибутивов еще нет официальных пакетов, но достаточно легко добавить PPA в Ubuntu (http://bit.ly/1uGO-lm6), загрузить пакеты AUR для Arch Linux (http://bit.ly/1o3ubTA) или воспользоваться любым шаманским способом, который предлагает ваш дистрибутив. Доступны также LXQt-версии Siduction (на основе Debian Unstable) и Manjaro. Есть даже экспериментальный релиз для Raspberry Pi, за которым стоит приглядывать.

В метапакете из Ubuntu PPA тоже есть несколько приложений: *JuffEd* (текстовый редактор) и удобный *QTerminal* с выпадающим списком. Хонг Чжен И [Hong Jen Yee] (PCMan) также портировал из LXDE файловый менеджер *PCManFM* и утилиту просмотра изображений *LXImage*.

Стоит отметить, что хотя релиз 0.7 считается стабильным, кое-где он ведет себя странно. Мы установили его в Ubuntu 14.04 и обнаружили, что наряду с добавлением своего собственного сеанса рабочего стола он также добавил такие пункты, как "Lubuntu Qt session" и "Lubuntu Nexus Session". Это побочный эффект слияния кода двух разных систем, и оба пункта не делают ничего особенно полезного. К счастью, пункт "LXQt Desktop" все-таки загружает нужный сеанс. Другой неясный момент в том, что проект мигрирует на *Qt5*, а в Arch Linux почти все пакеты AUR собраны для этой новой версии. К сожалению, поддержка *Qt5*, особенно вне KDE, слабая, и нам не удалось заставить тему QtCurve работать.



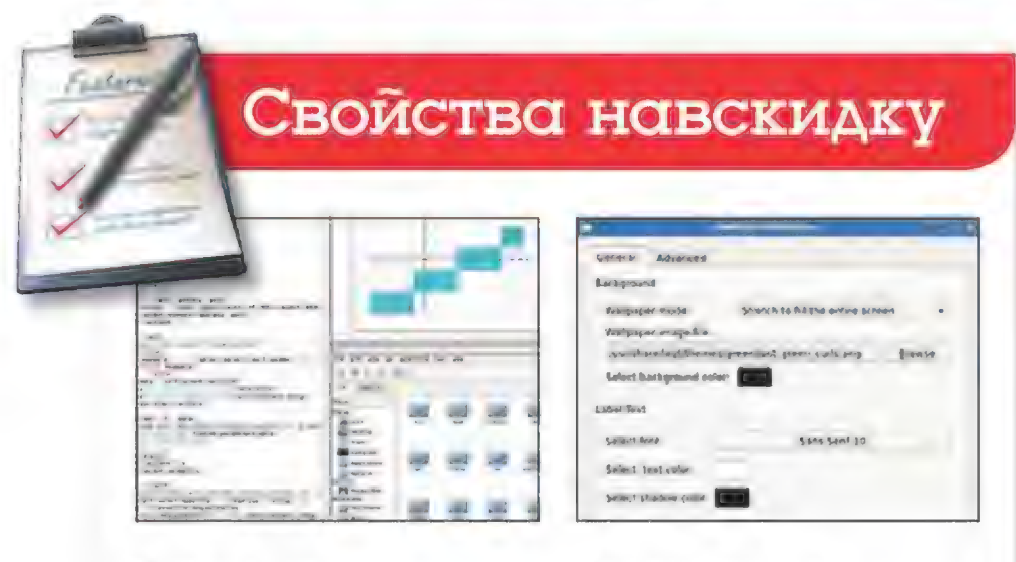
» **В LXQt есть свои панели управления, и с ними меню настроек Ubuntu кажется излишне переполненным. Но нам в Башнях LXF избыточность нравится...**

Обезжиренный рабочий стол

Если не считать нескольких косметических штучек, LXQt довольно легок. Менеджер окон по умолчанию *Openbox* полностью функционален, как и *Qt*-версия *PCManFM*, и все апплеты панели управления и виджеты панели задач. Так как рабочий стол легкий, сверх этого мало что есть, зато его можно дополнить множеством низкокалорийных приложений (см. «Дистрибутивы-малоежки», **LXF186**, стр. 22). По сравнению с «прародителями», этот дистрибутив по сути не предлагает ничего нового, даже по внешнему виду (хотя мог бы). Те, кто захочет его приукрасить, смогут легко заменить оконный менеджер (особенно если у них уже установлен *Metacity* или *kwin*), а панель выбрать поярче, наподобие Cairo Dock; но тогда дистрибутив утратит свою легкость.

Если вы не обладаете опытом погружения в тихую пустоту при работе без графической среды, вам легко упустить то, что происходит на самом деле: монтирование дисков, политики прав доступа, организацию меню приложений, привязку файлов и милые фоновые картинки. В LXQt все это просто работает, причем работает, потребляя менее 100 МБ оперативной памяти (согласно тестам, проведенным командой LXQt) — чем доказывается его эффективность.

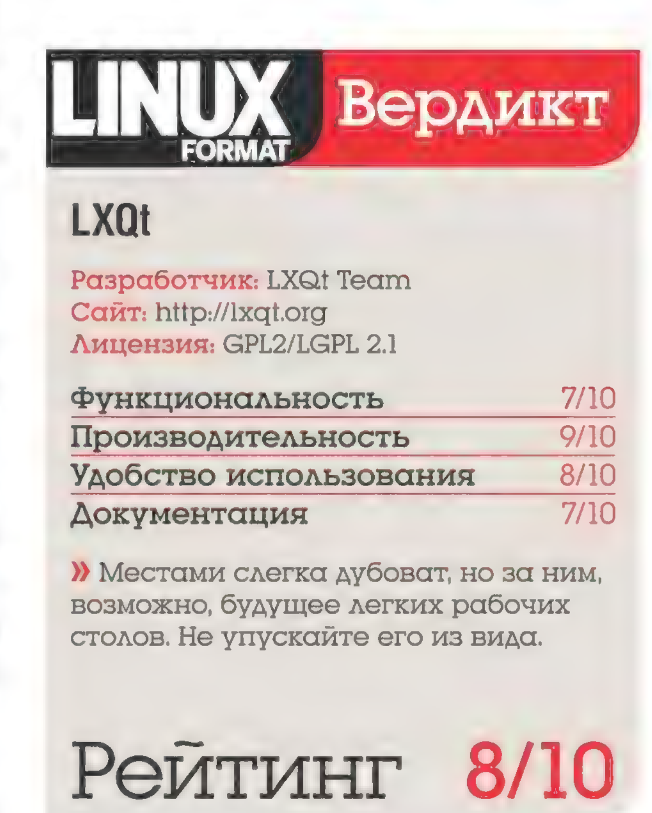
Qt гораздо более гибок, чем *GTK2+*, и гораздо приятнее на вид; нам особенно понравилось элегантное меню открытия файла, которое плавно изменяет размер иконок устройств на панели при перемещении разделителя. Это вполне стоит его перерасхода в 40 МБ памяти по сравнению с LXDE. LXF



Свойства навскидку

Приложения
JuffEd, *LXImage* и *PCManFM* позаботятся о ваших словах, картинках и файлах соответственно.

Без излишеств
Отсутствие миниатюр кое-кого может раздосадовать. Но окна работают нормально.



LINUX FORMAT Вердикт

LXQt

Разработчик: LXQt Team
Сайт: http://lxqt.org
Лицензия: GPL2/LGPL 2.1

Функциональность	7/10
Производительность	9/10
Удобство использования	8/10
Документация	7/10

» Местами слегка дубоват, но за ним, возможно, будущее легких рабочих столов. Не упускайте его из вида.

Рейтинг 8/10

OwnCloud 7

По выходу седьмой версии **Маянк Шарма** тестирует популярное открытое хранилище файлов и оценивает его позиции на фоне конкурентов.

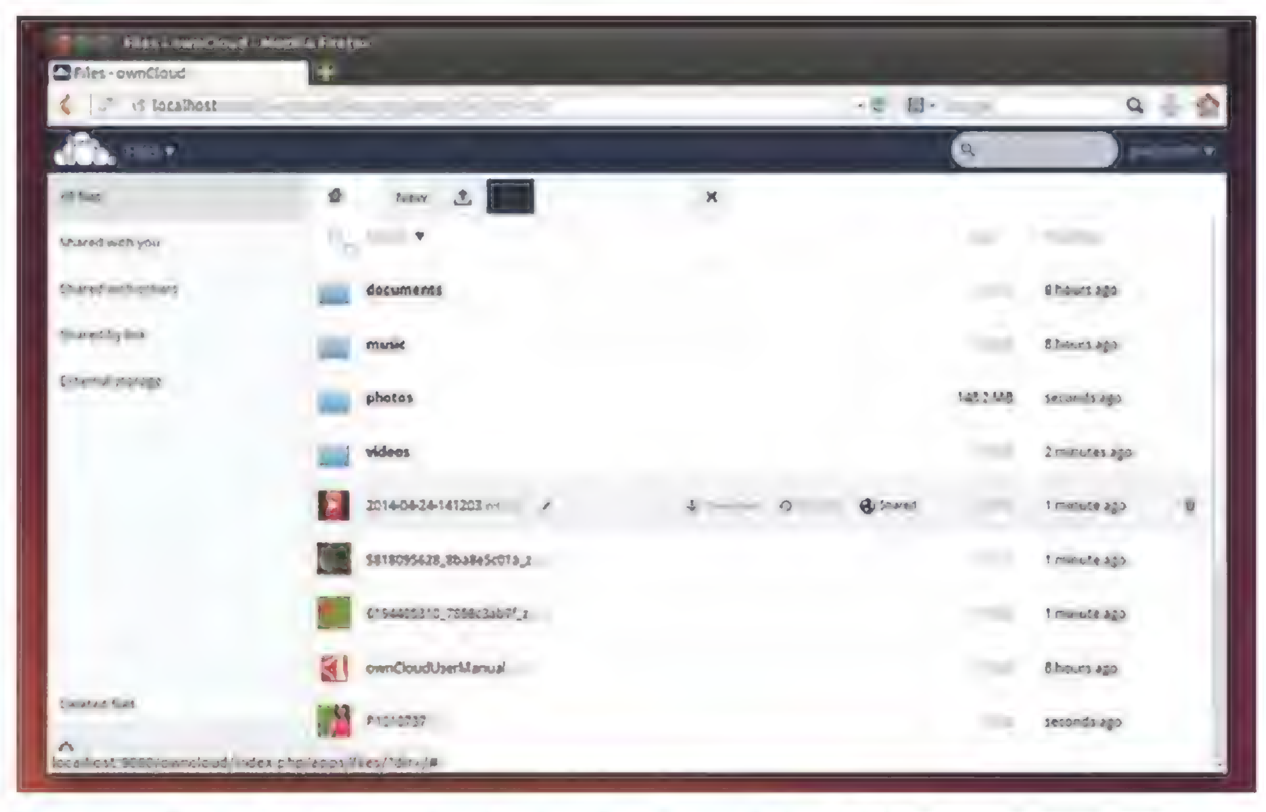
Вкратце

» Открытое облачное хранилище данных с возможностью синхронизации данных. См. также *Seafile* и *SparkleShare*.

Мы все стали чуть серьезнее относиться к передаче своих персональных данных в общий доступ, а благодаря решениям вроде *OwnCloud* процесс настройки автономного облачного сервера хранения данных намного упростился. Новое крупное обновление еще лучше отточено и еще удобнее в работе, чем все предыдущие.

Если вы пользовались онлайн-сервисами хранения файлов, то легко разберетесь с *OwnCloud 7 Community Edition*, даже если вы новичок. Прежним пользователям понравятся унифицированный и быстрый доступ к разным компонентам и приложениям. Панель управления теперь украшает информация в более удобной форме, включая отчеты о действиях над общими каталогами. Теперь можно попросить *OwnCloud* оповещать вас о новых действиях через электронную почту. Для подключения SMTP-сервера раньше приходилось редактировать файл настройки вручную, а в версии 7 это легко сделать через новый графический мастер на панели администрирования.

Работаете с множеством файлов? Сортируйте их по имени, времени изменения или размеру. Для ускорения выполняется «ленивая» загрузка файлов, т.е. по мере прокрутки страницы. Еще одно изменение, заметное при просмотре файлов — имя владельца файла; очень удобно при совместной работе. Ускорен механизм разделения файлов. Каталог для общих файлов по умолчанию **Shared** удален и заменен фильтрами общих файлов. Теперь общие файлы появляются в корневом каталоге, и их легко фильтровать по параметрам общего доступа. Аналогично реализовано новое представление со списками файлов на настраиваемых внешних ресурсах, например, на FTP, Dropbox, Google Drive и др.



› Интерфейс *OwnCloud* теперь можно просматривать на всех мобильных устройствах и планшетах и обращаться к нему даже с платформ без спецприложений, вроде Blackberry.

Крупнейшая новинка — межсерверное разделение файлов, т.е. возможность подключить друг к другу два экземпляра *OwnCloud*: скажем, смонтировать общий каталог от одного сервера *OwnCloud* в другой каталог на удаленном компьютере. Так, если вы создали общий ресурс в виде ссылки, то получатель, открыв эту ссылку, сможет добавить ресурс в свой экземпляр *OwnCloud*. Добавив ссылку, пользователи этого экземпляра *OwnCloud* смогут совместно работать с файлами и делиться ими.

Полегчало админам

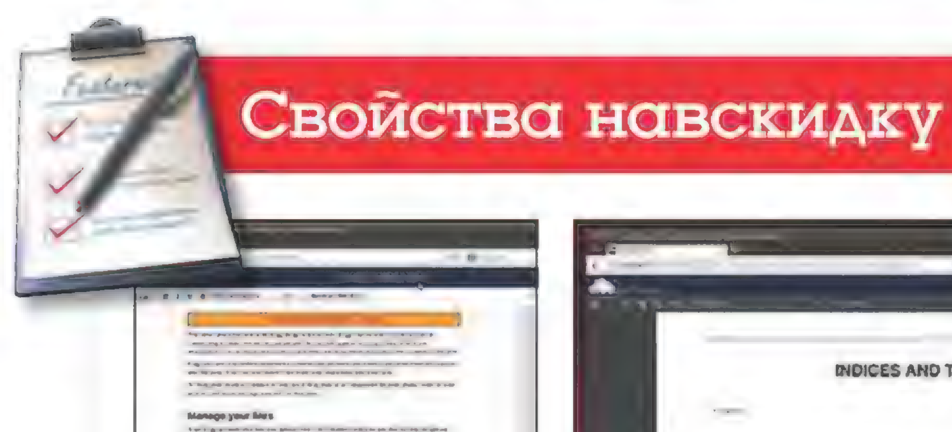
Яркая новая возможность — редактирование документов *MS Word*. При загрузке документов с проприетарными расширениями **.doc** и **.docx** приложение *Document* в *OwnCloud* конвертирует их на лету и редактирует наравне с файлами **.odt**.

По части общего доступа к файлам, в новой версии администраторам легче управлять поведением пользователей и обеспечивать безопасное размещение файлов. Отныне администратор может заставить пользователей задавать пароли при каждом создании ссылок на общие ресурсы или установить определенный срок действия ссылки в соответствии с сетевыми политиками безопасности.

Реформировано управление пользователями: администраторы получили более детальный контроль, включая указание допустимых типов хранилищ и приложений *OwnCloud* заданным группам пользователей. Процесс создания групп очень быстр и интуитивно понятен; к тому же стало можно создавать администраторов групп, управляющих учетными записями в назначенных им группах пользователей.

Благодаря внутренним переделкам очень ускорилась загрузка данных на сервер и с него, и снизилась нагрузка на сервер во время синхронизации файлов. Плюс, улучшились процедуры монтирования внешних хранилищ и версионирования файлов. Пускай в этом релизе не слишком много новых функций, но полно улушений производительности, значительно повысивших общее качество продукта. LXF

Свойства навскидку



Поддержка файлов MS Word

В приложении Documents наряду с .odt теперь можно открывать и редактировать файлы .doc и .docx.

Межсерверное взаимодействие

Стало можно совместно работать с пользователями в других экземплярах *OwnCloud*, через общие папки.



LINUX FORMAT Вердикт

OwnCloud 7

Разработчик: OwnCloud Inc
Сайт: www.owncloud.org
Лицензия: AGPL v3

Функциональность	8/10
Производительность	8/10
Удобство использования	8/10
Документация	8/10

» Отточенный релиз не только удовлетворит старых пользователей, но и привлечет новых.

Рейтинг 8/10

Dell PowerEdge T20

Присматриваете новый сервер? **Алан Стивенс** тестирует последний мини-сервер от Dell.

Вкратце

» PowerEdge T20 быстр и допускает установку большого количества жестких дисков. Базовый вариант довольно худосочен, возможности по управлению ограничены, но это вполне функциональная начальная система с хорошим соотношением цена/качество.

Новая малышка из серверного семейства Dell, PowerEdge T20, предназначена для использования дома и на небольших предприятиях в качестве платформы для хранения общих файлов и для размещения серверных приложений, таких как почтовые и web-серверы или серверы баз данных. T20 также сгодится как сервер отделения или филиала компании, где он столкнется с жесткой конкуренцией со своим главным соперником от HP в лице ProLiant MicroServer Gen 8.

Времена, когда можно было более или менее выделить свой сервер Dell из списка аналогов, давно прошли, особенно когда речь заходит о системах начального уровня вроде PowerEdge T20. Некоторых покупателей, возможно, разочарует (но упростит им жизнь) то, что единственный свободный слот в стойке T20 отдан источнику питания 290 Вт.

Корпус Dell, который ни в прочности, ни в компактности не сравнится с MicroServer Gen8 — удобная конструкция со съемной задней панелью, открывающей доступ к внутренностям сервера, включая компактную материнскую плату мини-ATX, установленную в направлении нижней части корпуса. В корпус можно установить до шести жестких дисков SATA, а металлическая конструкция позволяет разместить два диска размером 3,5 дюйма внизу и еще два — в верхней части корпуса.

К сожалению, диски нельзя менять «на лету», а для замены двух верхних дисков придется снять переднюю панель. Кроме того, хотя вместо оптического привода можно установить еще два 2,5-дюймовых жестких диска, у контроллера всего четыре порта SATA, и если вы захотите использовать все доступные отсеки для жестких дисков, придется устанавливать дополнительный адаптер.

Что касается самих дисков — можно либо поставить свои, либо подобрать готовые из традиционных жестких дисков объемом до 3 ТБ и SSD-диска на 160 ГБ. Встроенный RAID-контроллер поддерживает зеркалирование и чередование дисков, однако для RAID 5 вам (опять-таки) придется установить дополнительный контроллер.



› Сервер скромных размеров по скромной цене, с более чем скромным процессором.

Кроме этого, можно выбрать только процессор, и здесь вы ограничены двумя: Pentium G3220 с частотой 3 ГГц и Xeon E3-1225 v3 с частотой 3,2 ГГц. Экономные покупатели, скорее всего, выберут G3220, который со стандартными 4 ГБ памяти и без жестких дисков сохранит базовую стоимость сервера на отметке всего в £219 (около $375) без учета налогов. Но эта экономия может быть ложной, так как двуядерный G3220 — скорее настольный, чем серверный процессор, и большинство покупателей предпочтут ему обладающий бо́льшими возможностями Xeon.

При системе охлаждения на 84 Вт, 4 ГБ памяти и жестком диске SATA емкостью 1 ТБ эта конфигурация гораздо лучше подходит на роль сервера, но стоит всего £389 (около $670) без учета налогов, а если добавить второй диск 1 ТБ для зеркалирования, стоимость возрастет до £445 (около $765) без учета налогов.



Потенциал роста

Учитывая, что T20 — сервер, стандартные 4 ГБ ECC-защищенной памяти кажутся достаточно скупой цифрой, особенно потому, что добавить память не так уж дорого. К счастью, в базовом варианте 4 ГБ занят только один слот DIMM из четырех. Удвоить объем памяти до 8 ГБ можно всего за £50 (около $85) без учета налогов, и мы настоятельно рекомендуем это сделать, а при желании можете увеличить объем памяти до 32 ГБ.

Гигабитный сетевой интерфейс всего один, и некоторые покупатели захотят обзавестись дополнительными сетевыми адаптерами, а также более функциональным RAID-контроллером; для этого доступны три полномерных слота PCI Express (плюс один старомодный слот PCI).

Вы также повсюду обнаружите USB-порты — четыре спереди и шесть сзади. По два порта из каждого набора поддерживают USB 3.0; кроме того, в BIOS можно отключить ненужные порты, чтобы никому не удалось подсоединить к ним «левые» устройства.

Разумеется, сервер поддерживает Linux — по умолчанию на нем установлена Red Hat 6.5. Те, кому нужна привязка к производителю, могут выбрать продукт от Microsoft за £225 без налогов. Другой возможный вариант — загрузиться прямо в гипервизор с жесткого диска или SSD, но USB- или SD слота для встроенного гипервизора, как в HP MicroServer, здесь нет. LXF

Свойства навскидку



О корпусе

Несмотря на три кулера, T20 работает довольно тихо, и его вполне можно разместить в офисе.



Процессор Intel Xeon

При 4-ядерном процессоре, Xeon E3 T20 — очень проворный сервер и грузится за какие-то секунды.

LINUX FORMAT Вердикт

Dell PowerEdge T20

Разработчик: Dell
Сайт: www.dell.com/uk/business/p
Цена: £263+

Функциональность	7/10
Производительность	9/10
Удобство использования	7/10
Оправданность цены	9/10

» Многое от сервера по цене среднего настольного компьютера. PowerEdge T20 быстр и богат местом для установки добавочных дисков.

Рейтинг 8/10

Slice of Radio

Лес Паундер берется за беспроводное общение между Raspberry Pi и Arduino, но говорит ли он на том же языке?

Вкратце

» Платы расширения радиотрансивером Ciseco для плат Raspberry Pi и Arduino. Цель – беспроводное соединение множества плат в сети RF.

Спецификация

» Беспроводной диапазон 100–500 м
» Совместим со всеми платами RF Ciseco
» 128-битное шифрование
» Последовательное соединение между устройствами
» Беспроводное программирование устройств Arduino.

Беспроводная связь между Raspberry Pi и Arduino была возможна в прошлом через устройства линии Xbee, однако в Ciseco, где созданы различные радиочастотные (RF) устройства расширения для ПК, сделали нечто специально для Raspberry Pi и Arduino. Slice of Radio (SoR) — крошечная добавочная плата размером 3,4×3 см, созданная для Raspberry Pi моделей A и B (но совместимая с новым B+). Дальность связи — 100 м со встроенной антенной; внешняя антенна увеличивает ее до 500 м. Скорость передачи данных — около 250 Кбит/с, и их можно шифровать, используя 128-битный AES.

SoR — это трансивер, то есть устройство, способное и принимать, и передавать сигналы с устройств Ciseco. Компания также создала индивидуальную версию Raspbian OS под названием RasWIK. Эта ОС преднастроена для работы с SoR, но мы обнаружили, что ей нужна некоторая дополнительная настройка, в том числе программа *Minicom*.

Если кто пожелает добавить SoR к существующей установке Raspbian — можете настроить его специально для этого, а подробное руководство с инструкциями вы найдете на openmicros.org по адресу http://bit.ly/SliceGuide.

Чтобы дополнить SoR, Ciseco выпустила еще один радиотрансивер, щиток SRF, который работает для всех типов плат Arduino и других расширительных плат Ciseco RF. SRF можно запрограммировать в эфире с помощью Arduino IDE, позволяющего загрузить новый эскиз беспроводным способом в диапазоне 100 м при встроенной антенне-чипе или в диапазоне 1000 м с помощью внешней антенны.





› Stash предлагает страницы, важные для вас именно сейчас, в одном доступном для поиска перечне.

Обе платы поддерживают открытый протокол Lightweight Logical Application Protocol (LLAP). Он позволяет устройствам общаться по радио на простом языке, который работает с короткими «фразами», похожими на SMS, где перед отправляемым сообщением/считанной датчиком информацией указывается адрес.

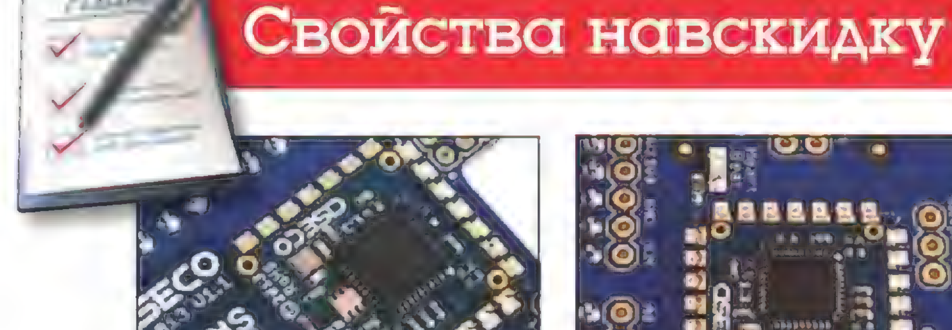
Грядущая прошивка

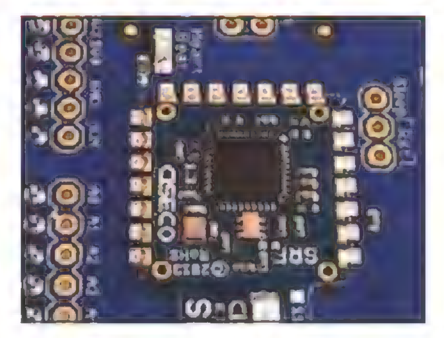
Мы протестировали обе платы на Raspberry Pi модели B, с RasWIK для SoR и Arduino Uno R2 для щитка SRF. С помощью RasWIK мы запустили среду Arduino IDE, рассчитывая быстро загрузить эскиз в SRF, но столкнулись с некоторыми проблемами загрузки. Хотя и не с первой попытки, мы сумели настроить SoR с помощью Minicom. Однако Ciseco сообщил нам о выходе более надежной прошивки (и, возможно, когда вы будете это читать, она уже появится).

Платы дарят проекту огромные возможности. Скажем, датчик температур, соединенный с вашим включенным через SRF Arduino, может передавать данные о температуре в помещении на вашу домашнюю систему отопления, работающую через Raspberry Pi, и если станет слишком холодно, включить центральное отопление. Однако обычно продукты Ciseco предназначены для энтузиастов — любителей работать с самыми передовыми проектами, и эти две платы — не исключение. Даже если взять преднастроенную ОС, SoR требует немалого времени, чтобы начать правильно работать со щитом SRF, и это может оттолкнуть часть пользователей. Однако обе платы исключительно хороши, если соединить их, и если вы — самоделкин, жаждущий обеспечить своему проекту возможности беспроводного соединения, то эти платы отлично вам подойдут.

Инструкции по настройке SoR на Raspberry Pi точны и хорошо написаны, однако придется немало прочитать и сделать, прежде чем устройство заработает. Скорейший способ установки и начала работы — использование RasWIK OS, и если ваш проект является мобильным, рекомендуем использовать прекомпилированный образ. Если вы используете B+ с SoR, Ciseco уверили меня, что они создают новый образ RasWIK, который позаботится о B+ и будет находиться в свободном доступе на их сайте. LXF

Свойства навскидку





Фантастический диапазон
Со встроенной антенной Slice of Radio способен на передачу в пределах прямой видимости до 100 м.

Трансиверы в обоих
Индивидуально адресуемые трансиверы означают, что можно перепрограммировать устройства прямо на ходу.

LINUX FORMAT Вердикт

Ciseco Slice of Radio

Разработчик: Ciseco
Сайт: http://shop.ciseco.co.uk
Цена: £3,90 (slice) + £9,99 (shield)

Функциональность	8/10
Производительность	8/10
Удобство использования	7/10
Оправданность цены	8/10

» Плата для программиста, который знает свое дело и хочет добавить функций в свой проект.

Рейтинг 8/10

Nvidia Shield

Последний планшет Android — сущий игровой титан. **Хью Лэнгли** и **Лили Прасьютсут** спорят, чья очередь играть на новом Shield.

Вкратце

» **Несомненно, Shield нашёл свою нишу. Может, он и не всем подойдёт, но геймерам уж точно. У него отличная цена и впечатляющая мощь.**

Спецификация

- » CPU: 4-ядерный Tegra K1 на 2,2 ГГц, ARM Cortex A15
- » ОЗУ: 2 ГБ
- » Хранилище: 16 ГБ + слот microSD
- » Экран: 8" IPS, 1920×1200
- » GPU: 192 шейдера Kepler
- » Порты: mUSB 2.0, mHDMI 1.4a
- » Беспроводная связь: 2×2 MIMO 2.4/5 ГГц 802.11n, GPS, Bluetooth 4.0 LE
- » Камера: 2×5 Мпикс (тыл и фронт)
- » Габариты, мм: 221×126×9,2
- » Вес: 390 г

Если оригинальная консоль Shield, странноватая помесь экрана и контроллера Nvidia, прошла мимо вас, вы не много потеряли. На момент релиза она казалась чуть недоделанной, потому-то, наверное, Nvidia не выпускала её за пределы США. Но затем в Nvidia вернулись к Shield. Наладив за это время обратную связь с пользователями, вняв их реальным запросам, они предложили несколько иной вариант: Shield Tablet.

Да, после всей этой шумихи насчёт слияния экрана с контроллером (эта модель теперь называется Shield Portable), в Nvidia снова их разделили. Причина? Nvidia считает, что предлагает нам лучшее из обоих миров: приличный планшет и крутое игровое устройство. Они пожелали создать первый в мире планшет консоль-класса, и, похоже, это им удалось. Пожалуй, еще более приятным фактором является цена: £240 для 16-ГБ модели, оснащённой только Wi-Fi, и £300 для варианта с LTE на 32 ГБ — довольно умеренно при таких характеристиках.

На первый взгляд планшет Shield выглядит как негабаритный HTC One. Но в ряду планшетов он где-то между Nexus 7 и Nexus 10. Он гораздо легче, чем ожидалось, и его удобно держать в руках, играя на сенсорном экране.

В плане характеристик, Shield Tablet имеет 8-дюймовый дисплей Full HD, 4-ядерный процессор ARM Cortex A15 Tegra K1 на 2,2 ГГц и 2 ГБ оперативной памяти. При потоковом воспроизведении через телевизор вы, вероятно, захотите подключить его к своим колонкам. Но в принципе есть пара динамиков на передней стороне планшета, дающих чистый и громкий звук.



› Планшет Shield создан для геймеров, но по сути это полноценный планшет Android.

Устройство также оснащено двумя камерами на 5 Мпикс — та, что с автофокусом HDR, расположена сзади, другая — фронтально. Если вы планируете использовать планшет Shield для игр на своем телевизоре, понадобится по крайней мере один контроллер, который обойдётся вам ещё в £50, делая устройство не столь малобюджетным, как хотелось бы. Дизайн аккумуляторного геймпада не претендует на инновационность, но весьма удобен: чувствуется стиль оригинального Shield. Большинство привычных кнопок на месте, но Nvidia добавила и несколько новых функций.

Шансы для пингвина

Три сенсорные кнопки Android образуют верхнюю V-образную секцию, а внизу теперь расположен тачпад, который также контролирует уровень громкости. И как в любом приличном игровом контроллере, к нему можно подключить гарнитуру. Nvidia даже добавила функции голосового управления, которая активируется удерживанием кнопки Home, когда вам захочется поспрошать Google.

Стоило ожидать, что для создания игрового планшета Nvidia пустит в дело тяжёлую артиллерию, и хотя вам пока не удастся запустить *Titanfall*, чип Tegra K1 — штука довольно мощная. Чтобы пошёл *Titanfall* и ему подобные, придётся дождаться, пока Nvidia закончит портирование GameStream под Linux для планшета и потокового ТВ. Nvidia также работает над переносом функциональности GeForce на Linux, включая автоматическое обновление драйверов и ShadowPlay, режима видеозаписи экрана с аппаратным ускорением, позволяющего в «фоновом режиме» записывать ваше прохождение игры для мгновенного обмена. Объединение потокового режима игры с Twitch и фронтальной камерой также является одной из главных фишек Shield, тем более вкупе с интеграцией ShadowPlay.

Как выяснилось, Shield особенно хорош в качестве игровой приставки в гостиной, даже без полной поддержки Linux. Благодаря возможности подключения планшета к телевизору для игр на большом экране и попарного соединения четырех контроллеров, запускаются игры Valve.

Shield может функционировать и как обычный планшет; но навряд ли многие, не будучи геймерами, станут покупать его, чтобы бродить по Интернету, сидя на диване; хотя и смотрится, и ощущается он весьма неплохо. LXF

Рост производительности

Тест (больше — лучше)	Nexus 7 (2013)	Старый Shield	Планшет Shield	Процентное изменение
Geek Bench	585	905	1,114	90,4/17,2
Geek Bench**	1,880	2,720	3,342	77,8/22,9
Antutu	20,330	38,063	42,551	109,3/11,8
Passmark	3,193	4,175	6,175	92,8/47,5
3DMark***	10,404	16,553	31,396	201,8/89,7
GFXbench T-Rex	892	1,962	3,116	249,3/58,82

* Одно ядро ** Многоядерный *** Ice Storm

LINUX FORMAT Вердикт

Nvidia Shield Tablet

Разработчик: Nvidia
Сайт: http://shield.nvidia.co.uk
Цена: £240

Функциональность	8/10
Производительность	9/10
Удобство использования	9/10
Оправданность цены	8/10

» Геймеры заслужили Nvidia Shield. Остаётся вопрос, потянет ли он Linux?

Рейтинг 9/10

X-Plane 10 Global

Напоминает *Top Gun*, но без волейбольной обстановки, считает **Дэвид Хейвард**.

Вкратце

» Последний образец серии *X-Plane* благополучно приземлился в Steam, но удастся ли ему переманить ярых поклонников *FlightGear*?

Несомненно, в списке авиатренажёров для пользователей Linux номер один занимает *FlightGear*. Эту замечательную программу ценят и энтузиасты, и любители. Но желающие не голого тренажёра, а более впечатляющей графики, достоверных звуков и реалистичной модели динамики полёта [Flight Dynamics Model, FDM] — выберут *X-Plane*.

X-Plane — не только потрясающий ландшафт. Это мощнейший и многогранный тренажёр, который учит вас летать, не ступая в кабину самолета. Улучшенная теория лопасти [Blade Element Theory] создает более точную FDM. Элемент лопасти принимает в расчет множество изменчивых параметров: силы, массу, гравитацию, ускорение, погодные условия, атмосферное давление, скорость ветра...



› Парите над миром в самолёте по своему выбору.

Как во всех хороших имитаторах, условия полета можно менять на свой вкус. Хотите полетать на Cessna в окрестностях Сиэттла — пожалуйста. Мечтаете вылететь из аэропорта в Норвике на экспериментальном X-15, или даже на космическом шаттле — почему бы, чёрт возьми, и нет?

Но пусть взмыть в небеса из Хитроу на Локхид F-117 «Найтхок» звучит волшебно — с этим есть и проблемы. Хотя имеется доступ к континентальным картам DLC, многие районы ещё не разведаны. Никто не ждёт от *X-Plane* точности *Google Maps*, но явно не помешало бы что-то кроме вида аэростатов и водных просторов. Стабильность также под вопросом. Иногда вместо земли вдруг видишь небо; здания то возникают, то пропадают; да и приборная панель в кабине не без огрехов графики. Помимо всего, что связано с видеорядом, *X-Plane* у нас ещё и несколько раз падал.

Невзирая на эти недостатки, *X-Plane* — не просто игра, а весьма впечатляющий имитатор: здесь легко наматывать часы полётного времени. В полной мере оценить его по достоинству можно только при наличии хороших практических знаний в воздухоплавании: многие тонкости новичкам и невдомёк. LXF



LINUX FORMAT Вердикт

X-Plane 10 Global 64-bit

Разработчик: Laminar Research
Сайт: http://bit.ly/X-Plane10Global
Цена: £45

Игровой процесс	9/10
Графика	6/10
Продолжительность	5/10
Оправданность цены	5/10

» Ну очень достоверный имитатор полёта; дороговат, особенно на фоне FlightGear, но во многом лучше.

Рейтинг 7/10

Blockstorm

Дэвид Хейвард перевоплощается в воксельного Рэмбо.

Вкратце

» Воксельный стиль напомнит многим *Minecraft*, только с кучей пушек и взрывов.

Blokstorm — стрелялка от первого лица, действие которой разворачивается в блочной пиксельной среде в стиле *Minecraft*.

В *Blockstorm* три игровых режима, и каждый матч состоит из пятиминутных сессий. У вас широкий выбор персонажей, в том числе спецназовец, солдат пустыни, снайпер, шеф-повар и даже некто похожий на классического Человека-волка Лона Чейни-младшего [Lon Chaney Jr.]. С помощью редактора вы сможете создать и собственного, уникального игрока.

Предложенный арсенал оружия напоминает скорее *Counter-Strike*, чем безумные пушки и заколдованные мечи *Minecraft*. К вашим услугам разнообразные пушки, плюс лопата, чтобы вскапывать грунт, блок для строительства и нож для рукопашного боя.

Blockstorm имеет одну притягательную особенность, выделяющую его среди прочих FPS-игр: его среда полностью разрушаема, что добавляет в игровой процесс «новые повороты», которые разработчики стремятся совершенствовать (а журналисты, пишущие об играх, обожают описывать). Выбивать почву из-под ног противников — занятие приятное, равно как и крушить опоры громадной конструкции и смотреть, как она падает на головы незадачливых игроков. Однако будьте осторожнее, ведь то же самое может случиться — и непременно случится — и с вами.

Игровой редактор позволяет построить целый мир, так что у вас есть всё, чтобы создавать какие-нибудь мудрёные арены для сражений.

Blockstorm — развивающаяся игра, и разработчики призывают игроков вносить свой вклад в её становление. Это амбициозный, но увлекательный проект, его плюсы — простота и командный дух. И всё же, до *Minecraft* ей ещё далеко. LXF



› *Blockstorm* становится действительно оригинальным, когда действие закручивается.



LINUX FORMAT Вердикт

Blockstorm

Разработчик: GhostShark Games
Сайт: www.playblockstorm.com
Цена: £9

Игровой процесс	7/10
Графика	7/10
Продолжительность	8/10
Оправданность цены	8/10

» Весёлая и увлекательная стрелялка от первого лица с воксельной графикой и обстановкой в духе Minecraft.

Рейтинг 7/10

Wasteland 2

Евгений Балдин играет в *Wasteland 2* и вспоминает детство, частично проведённое на пустошах *Fallout* с энергетической винтовкой наперевес.

Вкратце

» Ролевая игра в фантастическом пост-апокалиптическом мире с пошаговыми тактическими боями, морем текста, нетривиальными людскими (и не только) сообществами и совершенно не обязательными квестами.

Это оно! Та самая «старая школа» ролевых игр. Ну-у, не совсем старых, примером которых и был *Wasteland 1*, но тех старых, к которым относятся незабываемые шедевры типа *Fallout 1* и *2*, *Baldur Gates*, *Planescape Torment* и прочее. Брайан Фарго [Brian Fargo], приложивший в своё время руку ко многим из игр «старой школы», два года назад собрал на Kickstarter три миллиона долларов под обещание сделать ролевую игру в пост-апокалиптическом сеттинге в духе *Fallout*, но не *Fallout*, потому что это *Wasteland 2*. И это обещание было выполнено! Да, есть огрехи (за менее чем месяц после релиза вышло два весьма объёмных патча); да, скажем так, маловато портретов для неигровых персонажей (по-хорошему, надо бы увеличить их разнообразие хотя бы на порядок — возможно, это будет исправлено сообществом в будущем); да, модельки персонажей весьма посредственные и выглядят не очень красиво (а много вы видели красивых людей и ярких одежд среди живущих в мире после тотальной ядерной войны?); да, с локализацией есть проблемы (хотя в принципе к переводу и шрифту можно привыкнуть, проблемы же с прохождением основной ветки приключений вроде как исправлены в последнем патче); да, из игры выкинули возможность скрытого прохождения (весьма печально); да, это не *Fallout* (это лучше!); но ощущения при путешествии по пустыням Аризоны и в особенности по Лос-Анджелесу вполне себе аутентичны тем самым детским ощущениям при путешествии по Южной Калифорнии в роли Избранного.



› Вообще-то в игре не нужно делать ничего, кроме розыска и починки радиовышек в целях пеленгации источника странных радиопередач с угрозами. Это одна из них, уже очищенная от гигантских растений-мутантов.

Поддержка GNU/Linux была гарантирована не сразу по началу кампании на Kickstarter, но, по всей видимости, стихийный наплыв любителей «старой школы» из сообщества заставил обратить внимание на нашу основную платформу. Возможно, эта компания и была определяющей в скором портировании популярного игрового движка Unity 3D, используемого в *Wasteland 2*, на GNU/Linux. С портом у меня не было никаких проблем — всё работало стабильно; правда, следует отметить, что на «игровом» домашнем компьютере у меня стоят Ubuntu 14.04 и карта Nvidia GeForce GTX 680 с закрытыми двоичными драйверами, и я отключаю вспомогательный монитор через nvidia-settings перед запуском steam-версии игры.

Игра разбита на две крупные непересекающиеся карты: Аризона и Лос-Анджелес. В отличие от пустынной Аризоны, Лос-Анджелес гораздо более богат побочными приключениями и интересными историями; правда, и багов с прохождением этих квестов там встречается поболее в силу того, что в открытом тестировании бета-версии присутствовала только карта Аризоны. Необходимо отметить, что более 90% квестов *Wasteland 2* не являются обязательными. По сути дела, наследникам солдат инженерного батальона из *Wasteland 1* нужно только найти и подсоединить к сети шесть радиовышек, да в случае Лос-Анджелеса дополнительно придётся отыскать и принести сержанту Вудсону несколько десятков килограммов наполнителя для кошачьих туалетов. Как бы вам ни втолковывали, что Пустынные рейнджеры [Desert Rangers] обязаны делать мир лучше, делать это совершенно не обязательно, хотя и интересно. Мир *Wasteland 2* населяют около двух десятков фракций — от почти нормальных на прочем фоне бандитов и зелёных классических зомбиобразных мутантов до вежливых человечков в силовой броне и монахов с ядерными рюкзаками за спиной. Кого из них поддержать,

Свойства навскидку



Старая добрая фантастика

RPG с пошаговыми тактическими боями в мире после ядерного взаимного уничтожения. Огромные локации для исследования, куча «нычек» для откапывания, сейфов для вскрытия и барахла, которое можно поднять. Мир способен спасти только ядерный взрыв — что же ещё?



Нет нужды быть добрым

...или честным, хотя это тоже интересно. Интересная вселенная с массой любопытных, странных и извращённых людских и не очень фракций, каждая из которых хочет что-то своё. Оставить их вариться в собственном соку или с болью и криками вытащить их на путь цивилизации — решать вам.



› Бои тут пошаговые. Казалось бы, какие-то барсуки-переростки, но пока отряд находится в начале пути, приходится закидывать их гранатами.

а кого отправить на тот свет — решать вам. Если после знакомства с вами неигровые персонажи остались в живых и почему-то прониклись к вам симпатией, то они придут и помогут в последней битве со Злом, которому вы вполне можете сдаться, а в ответ за услуги могут подкинуть координаты «странных мест» со странными структурами, которые одаривают членов отряда опытом или очками навыков. Помощники из них так себе, но получается весьма мило.

Как и подобает для классической компьютерной ролевой игры, начинается всё с создания команды. По сравнению с общим временем прохождения это займёт относительно немного времени (120 часов на всё про всё), но решение — точнее, компромисс, принятый в начале — будет аукаться в течение всего прохождения.

В отличие от *Fallout*, тут нужно создать команду из четырёх бойцов. Один рейнджер в поле не воин. У каждого есть возможность изменить основные параметры и выбрать полезные навыки.

Основные параметры — это Координация [Coordination], Удача [Luck], Восприятие [Awareness], Сила [Strength], Скорость [Speed], Интеллект [Intelligence] и Харизма [Charisma]. Каждые десять уровней можно увеличить какой-либо из основных параметров на 1. Для рейнджера потолок равен 50 уровней; реально ближе к концу набрать чуть более 40 из них. Уже на основе этих параметров вычисляются характеристики персонажей. Кроме привычных Очков действия [Action Points], для бойца ключевой характеристикой является Инициатива [Combat Initiative], которая определяет очередь действия во время боя. Для ориентированного на бой рейнджера инициатива должна быть около или больше 12, иначе противник будет стрелять и бить раньше него. Если вы планируете создать специалиста по ближнему бою, то следует обратить внимание и на Скорость перемещения [Combat Speed], множитель для AP.

В отличие от того же *Fallout*, где энергетическое оружие было улучшенной версией стрелкового, в *Wasteland 2* энергетическое оружие хорошо работает только на бронированных противниках. Если для эффективного поражения в случае обычного оружия параметр брони жертвы должен быть ниже пробивной способности оружия, то для энергетического оружия всё наоборот. Иными словами, если вооружить всех исключительно энергетическим оружием, то команду «запинают» бомжи в майках, а вот бронированные роботы будут умирать на ура.

Умение работать с оружием или ввернуть верную фразу в процессе разговора целиком определяют навыки. Их много, и сразу всё выучить не получится. На скорость изучения навыков влияет Интеллект. По возможности он не должен быть ниже 4, что соответствует трём очкам навыка на уровень. Чтобы изучить ту или иную ветку полностью, нужно 44 очка.

Кроме основной команды, в процессе путешествия можно присоединить до трёх игровых персонажей. Сразу же на локации Цитадели рейнджеров игроку даётся возможность взять в команду высокоуровневую «няньку», ветерана *Wasteland* Анджелу Дет [Angela Deth]. Следует, однако, понимать, что такие подарки делаются на время.

> «Прелесть игрового мира проявляется в приятных мелочах.»

Есть очень много способов распределить роли среди своей команды. Ниже я предлагаю своё, наверняка неоптимальное решение. В команде должен быть:

1 Лидер (Co:2, Lu:1, Aw:1, St:3, Sp:1, In:10, Ch:10) Упор делается на Лидерство, Взлом замков, Взлом сейфов, Взрывотехнику и немного на Зоркий глаз [Perception]. Это абсолютно не боевой персонаж, и его цель — помогать сопартийцам своим лидерством и заниматься взломом и разминированием. В качестве оружия оптимально энергетическое.

2 Снайпер (Co:7, Lu:1, Aw:10, St:3, Sp:2, In:4, Ch:1) Упор на снайперскую винтовку. Позже имеет смысл изучить немного выживания до 5 и энергетическое оружие. Вначале это основной боец отряда, в силу высокой инициативы.

3 Рукопашник (Co:2, Lu:1, Aw:3, St:10, Sp:7, In:4, Ch:1) Упор на Холодное оружие, Полевую медицину и Оружейное мастерство [Weaponsmithing]. В силу высокой скорости этот боец силовой поддержки успеет дойти и вылечить раненого. Оружейное мастерство является весьма важным навыком, позволяющим разбирать оружие на составляющие. В Цитадели рейнджеров куски разобранного оружия можно продать дороже, чем само оружие. Кроме того, апгрейд оружия позволяет существенно повысить его дальность и точность стрельбы.

4 Болтун (Co:4, Lu:1, Aw:5, St:3, Sp:4, In:10, Ch:1) Упор на способности Хитрожопый, Жополиз, (да, да, мне тоже не нравится такой буквальный перевод Smart Ass и Kiss Ass) и Штурмовые винтовки. Позже нужно будет изучить навык Задира [Hard Ass].

В качестве спутников я выбрал Розу [Rose], Пизепи Йорен [Pizepi Joren] и Зубило [Chisel]. Роза обладает высоким интеллектом, что позволяет свалить на неё навыки Компьютеры и Хирургия, а позже — Ремонт тостеров и Механика. Это крайне полезный персонаж, и чтобы присоединить её, нужно спасти сначала СХ Центр [AG Center]. Пизепи Йорен обладает навыком Энергетического оружия и хорошим Зорким глазом (для присоединения необходимо, чтобы суммарная харизма команды была не меньше 21). Зубило же довольно сильный рукопашник.

Прелесть игрового мира проявляется во множестве совершенно не обязательных, но приятных мелочей. Отдельно можно долго хвалить звуковое оформление. Музыка ненавязчиво создаёт фон пост-апокалиптического мироощущения, а если на карте Аризоны взять чуть западнее Цитадели рейнджеров и пойти на север, то можно будет поймать радиопередачу, где на чистом русском языке «Чебурашка вызывает Шапокляк»... Я выставил 10 из 10, потому, что это так и есть, несмотря на простенькие модельки персонажей и нехватку портретов. LXF



Вердикт

Wasteland 2

Разработчик: InXile Entertainment
Сайт: www.inxile-entertainment.com
Цена: 599 руб. на Steam или gog (DRM-free версия)

Сюжет	10/10
Графика	6/10
Увлекательность	10/10
Оправданность цены	10/10

» Фантастичная ролевка «старой» школы. Мелочи, создающие атмосферу, завораживают, и про огрехи забываешь.

Рейтинг 10/10

ИЗДЕРЖКИ ПРОПРИЕТАРНОСТИ

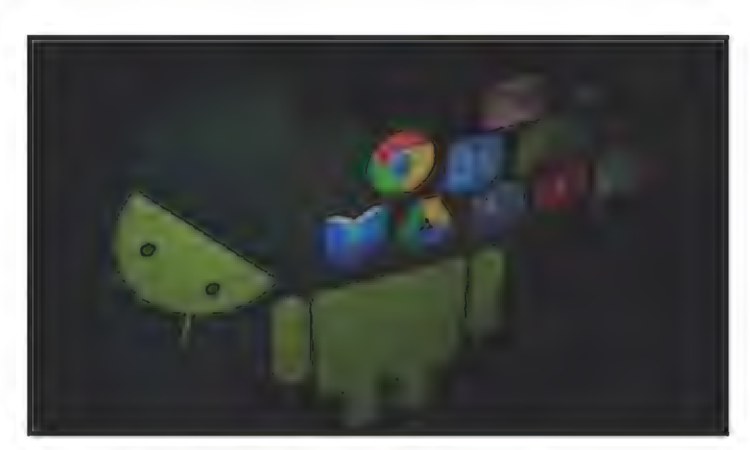
Google разбушевался

Начался диктат производителям Android-устройств.

Android, открытая ОС от Google для мобильных устройств, становится всё менее открытой: Google стремится усилить контроль. Растет число приложений от Google, типа Gmail, Drive, *Chrome*, обязательных к предустановке на новом устройстве: например, иконки Google Play Store и папки с основными приложениями Google должны быть на домашнем экране на самом виду и сопровождаться виджетом Google Search, а «перелистывание» домашнего экрана вверх — вызывать Google Search, как на устройствах Nexus.

Производители устройств давно рекламируют Android как наиболее «открытую» и настраиваемую ОС. А доходы Google идут не от продажи устройств с Android, а от потребителей, применяющих Google Search и другие продукты компании. Google хочет упорядочить многообразие интерфейсов ОС Android и сторонних приложений для неё, наличие которых отодвигает на задний план сервисы и продукты компании, явно принуждая производителей ставить на телефоны близкую к стоковой версию Android, с приложениями Google.

Производители хором характеризуют новые правила Google как «закручивание гаек», но те идут и на немалую пользу потребителям. Так, Samsung снабжает свои Android-устройства тяжеловесными интерфейсами со своими приложениями, порой менее стабильными или ничем не лучше аналогов от Google. Да и они просто иные, что мешает пользователям разных Android-устройств помогать друг другу в настройке и способно вызвать отток пользователей на iPhone, с более дружелюбным интерфейсом. Google также пытается заинтересовать производителей устройств, предлагая им часть дохода от своих сервисов (Google Search или Google Play) при отказе от предустановки продуктов конкурентов вроде Yahoo или Bing.



› **Багаж зелёного человечка становится всё больше и больше....**

Правила касаются только устройств с сервисами Google, включая самый важный — магазин Play Store. Условия же проекта AOSP (Android Open Source Project) позволяют производителям оборудования модернизировать устанавливаемую ОС, отказавшись от многих сервисов Google; яркий пример такого подхода — Amazon Kindle.

НА ВАШЕМ ТЕЛЕФОНЕ

Android ведет и везет

Мобильные приложения работают на горожан.

Российские разработчики обновили версии своих приложений для Android.

В приложении 2ГИС для Android теперь отображаются планы этажей торговых центров. Так, в Новосибирске новый сервис функционирует в «Ауре», «Меге» и «Версале»; в Москве 2ГИС покажет планы этажей торговых центров «Метрополис», «Мега Тёплый Стан», «Мега Химки», «Vegas», «Афимолл Сити», «Город» (Лефортово), «Мега Белая Дача», «Европейский», «Золотой Вавилон». В целом, в России доступны планы этажей более 30 моллов, и список в будет расширяться. При этом объекты инфраструктуры каждого этажа обозначены специальными значками, что значительно упрощает навигацию.

2ГИС — российская ИТ-компания, выпускающая одноимённые электронные справочники с картами городов. «Ориентироваться в городе с помощью мобильного устройства — это быстро, удобно и привычно. Мы хотим перенести этот опыт в торговые центры, — комментирует Елена Толкачева, директор департамента производства 2ГИС. — И мы рады, что руководство ТЦ позитивно относится к сотрудничеству». В дальнейших планах 2ГИС — выпуск многоуровневых карт и для других объектов с большим потоком посетителей.



› **Вы не мудрите — покажите пальцем! Такси доставит вас в любую точку на карте.**

Обновлена версия приложения Яндекс.Такси для Android. Новый интерфейс разработан таким образом, чтобы всё делалось последовательно и быстро — без лишних переходов, говорится в блоге компании. Приложение быстро найдёт свободную машину рядом с вами, на выбор из классов «эконом», «бизнес» или «комфорт». Среднее время подачи — 7 минут. Вызвать такси сейчас можно в Москве, Санкт-Петербурге и Воронеже.

При заказе, «Яндекс.Такси» определяет местоположение пользователя по GPS, показывая это место на карте. Уточнение адреса осуществляется перемещением карты в нужную точку. Сервис запоминает параметры последнего заказа и автоматически предлагает их при следующем запуске.

За последний год в Москве сервисом «Яндекс.Такси» воспользовались около 670 тыс. пользователей, совершивших примерно 6 млн заказов; 80% пассажиров вызывали такси на ближайшее возможное время — в течение 10 минут. На связи с «Яндекс.Такси» в Москве 8–10 тыс. водителей, и зачастую такси приезжает раньше. Теперь, если заказать машину просто на ближайшее время, она приедет в кратчайшие сроки. Можно зарезервировать такси и на любое удобное время.

БОЙТЕСЬ ТРОЯНЦЕВ

«Люди! Будьте бдительны!»

Приложения для Android отнюдь не безопасны.

Instagram, Grindr, OkCupid и многие другие популярные приложения для Android не соблюдают элементарных мер по защите данных, ставя под угрозу конфиденциальность своих пользователей. Таков вывод Группы обучения и расследований компьютерных инцидентов Университета Нью-Хейвена (UNHcFREG), ранее обнаружившей уязвимости в чат-приложениях WhatsApp и Viber. На сей раз целью их анализа был поиск слабых мест, ведущих к перехвату данных. На своём канале YouTube (https://www.youtube.com/channel/UCxdY4Hew6gdGbIN6dV2eJbg) исследователи выложили видео о своих результатах, затрагивающих свыше 1 млрд пользователей. «Разработчики приложений очень неаккуратны», отметил Ибрагим Баггили [Ibrahim Baggili], директор UNHcFREG и главный редактор журнала "Journal of Digital Forensics, Security and Law".

При помощи инструментов анализа трафика, таких как *Wireshark* и *NetworkMiner*, исследователи выявили, где приложения хранят данные и как их передают. Например, Instagram до сих пор хранит изображения на серверах в незашифрованном и общедоступном виде. То же касается и приложений OoVoo, MessageMe, Tango, Grindr, HeyWire и TextPlus, при отправке фоток от одного пользователя другому. Эти сервисы хранили контент как простые ссылки "http", далее направляемые получателям. Но доступ к такой ссылке — по сути, и доступ к изображению, поскольку аутентификации не предусмотрено. По мнению г-на Баггили, сервисы должны либо быстро удалять изображений со своих серверов, либо выдавать к ним доступ только аутентифицируемым пользователям.

> «Владельцам мобильных устройств следует быть осторожнее.»

Ряд приложений, в т.ч. OoVoo, Kik, Nimbuzz и MeetMe, не шифруют журналы чата на устройстве, и завладевший устройством злоумышленник может сбросить дамп и увидеть сообщения. А другие приложения не шифруют логи чатов на сервере. Многие приложения либо не используют протокол SSL/TLS (Secure Sockets Layer/Transport Security Layer), включая применение цифровых сертификатов для шифрования трафика, либо используют его небезопасно. Хакеры могут перехватить незашифрованный трафик по Wi-Fi, если жертва находится в общественном месте — так называемая атака «человек-посредник», MITM-атака (*англ.* Man in the middle). Протокол SSL/TLS считается основной мерой обеспечения безопасности, хотя и не на 100% действенной.

Около 3 млн человек используют приложение OkCupid, не шифрующее чаты через SSL. Видео демонстрирует, как при помощи сниффера трафика увидеть отправленный текст, а также и самого отправителя. Исследователи пытались связаться с разработчиками приложений, но во многих случаях им это не удалось. На обращения в техподдержку по адресам электронной почты ответов зачастую также не последовало.

Многие современные вирусы для мобильных устройств незаконно собирают информацию и шпионят за пользователями. Недавно обнаружен именно такой Android-троянец; причём его атака рассчитана на конкретную группу лиц. Шпион ставился на мобильники протестующих жителей Гонконга, выступающих за более демократичную избирательную систему, под видом приложения для координации их действий, не вызвав подозрений у большинства своих жертв. Эксперты компании «Доктор Веб» исследовали его, дав ему имя *Android.SpyHK.1.origin*. Он ищет GPS-координаты устройства, эксплуатируя уязвимость системного виджета управления питанием, позволяющую активировать некоторые функции в обход глобальных системных настроек. В ряде случаев троянец теоретически может активировать GPS-приемник, даже если владелец устройства его отключил. Мало того, шпион обеспечивает прослушку в реальном времени, передавая диктофонную запись на сокет удаленного управляющего сервера. Это неплохая альтернатива прослушке через скрытый телефонный звонок: если каналы сотовых сетей блокированы правоохранительными органами, вероятность доступных Wi-Fi-сетей остается весьма высокой, и у злоумышленников есть шансы добыть нужные им сведения. При достаточно мощном сервере, объединив зараженные смартфоны и планшеты в систему слежения, можно получать оперативную информацию об обстановке, а также и возможных действиях в будущем. Не исключено применение этой или аналогичных вредоносных программ и в других регионах мира; владельцам мобильных устройств следует быть осторожнее и не устанавливать сомнительные приложения.

На заре становления рынка ПК большинство вирусов создавалось ради развлечения, бахвальства, доказательства своих навыков программиста, выражения своей точки зрения на события либо с целью развлечения или хулиганства, а не для материальной выгоды; но постепенно интерес вирусописателей дрейфовал в сторону незаконного заработка, и сейчас сложно встретить троянца, созданного с иной целью. Тем интереснее было вирусным аналитикам компании



› **Такую визитную карточку оставляет в телефоне наглый вирус.**

«Доктор Веб» получить в свое распоряжение еще одну необычную угрозу — из редкого класса программ-вандалов. Ценный академически, новый Android-троянец, внесенный в вирусную базу компании как *Android.Elite.1.origin*, весьма опасен, т.к. удаляет все имеющиеся на карте памяти данные, мешает читать входящие SMS-сообщения и общаться в популярных программах-мессенджерах. Распространяется он под видом популярных приложений, например, игр. При запуске *Android.Elite.1.origin* обманным путем пытается получить доступ к функциям администратора мобильного устройства, якобы нужный для завершения корректной установки приложения. При попытке пользователя запустить клиент Facebook, программу WhatsApp Messenger, Hangouts либо стандартное системное приложение для работы с СМС-сообщениями вирус блокирует их активное окно, выводя на экран изображение с текстом OBEY or Be HACKED [Повинуйся, или будешь взломан]. Данная блокировка сохраняется только для указанных программ, но не для прочих приложений или ОС в целом. Кроме того, *Android.Elite.1.origin* каждые 5 секунд рассылает по всем контактам телефонной книге SMS "HEY!!! <имя контакта> Elite has hacked you. Obey or be hacked."

Похожий текст отправляется в ответ на все входящие SMS с мобильников других пользователей — счет мобильного телефона большинства пострадавших опустошается моментально. LXF

В данном выпуске новостей использованы материалы с сайтов https://www.theinformation.com, http://info.2gis.ru, https://mobile.yandex.ru, http://www.computerworld.com, http://news.drweb.com

Linux в телефоне

Маянк Шарма покажет, как расширить возможности смартфона, установив на него полноценный дистрибутив Linux.



Наш эксперт

Маянк Шарма не работал сисадмином, но считает, что после стольких лет установки и настройки Linux на куче машин для самых разных целей он знает своё дело.

Ваш телефон Android уже набит благами Linux, но вы можете обогатить его ещё больше и найти лучшее применение многоядерным процессорам и уйме оперативки, установив полноценный дистрибутив Linux параллельно с мобильной ОС.

Есть масса причин иметь у себя в кармане хороший Linux-ПК. Например, создать на нём сервер LAMP запуска web-приложений и обслуживания web-страниц. Если вы сетевой администратор, установите свои любимые инструменты Linux и превратите смартфон в портативный инструмент устранения сетевых отказов или средство тестирования на проникновение.

Проект LinuxOnAndroid предлагает Android-приложение и несколько скриптов оболочки, а также кучу дистрибутивов Linux, загружаемых приложением. Скрипты в два счёта смонтируют образ Linux на файловой системе Android и SD-карте внутри файловой системы Linux, затем запустят *Chroot*, чтобы изменить корневой каталог на Linux и открыть оболочку для работы со смонтированной системой. Скрипты также настроят SSH для защищенного удаленного доступа и VNC для доступа к графическому рабочему столу.

Скрипты обеспечивают только подготовку для запуска Linux на устройстве. Файлы образов Linux, предоставляемые данным проектом — просто окружения Linux, упакованные разработчиками проекта и настроенные с учетом различных сценариев использования и аппаратных возможностей.

> «Есть масса причин иметь у себя в кармане хороший Linux-ПК.»

Важно уточнить, что все эти дистрибутивы Linux состоят из пакетов ARM и предназначены не для виртуальных машин, а для реального оборудования на вашем смартфоне Android. И поэтому подойдут даже для самых скромных устройств. Мы, например, ухитрились запустить Arch Linux с рабочим столом *Enlightenment* на телефоне Samsung с одноядерным процессором на 1 ГГц и 400 МБ оперативной памяти.

На момент написания статьи, в составе проекта — стабильные образы Arch Linux, Debian Lenny, Debian Testing, Fedora 20, Fedora 19, Kali Linux, Ubuntu 13.10, Ubuntu 13.04 и Ubuntu 12.04 LTS. А также не проверенные в эксплуатации и потому не предназначенные для производства других дистрибутивов Alpha-образы, в том числе Slackware, Bodhi, openSUSE и Ubuntu 14.04. Причем каждый доступен в нескольких редакциях.

Образ Core [Загрузочный] — самый маленький, он содержит минимальный набор пакетов, необходимых для запуска дистрибутива. Он лучше всего подходит для создания собственных образов и включает оконный менеджер *Openbox*. Далее идёт Small [Малый] образ, в его составе — рабочий стол *LXDE* с пакетом программ. Наконец, есть Large [Большой] образ, включающий KDE Plasma Desktop, *LibreOffice* и *GIMP*.

Некоторые дистрибутивы имеют и дополнительные образы. Так, в Arch Linux есть образ рабочего стола *Enlightenment*, а в Fedora — рабочего стола *Mate*. Доступные варианты файловых систем — ext2 или ext4. Те образы, что на ext4, совместимы с Android 4.3, а те, что на ext2 — с более ранними версиями Android.

Скорая помощь

LinuxOnAndroid — полностью открытый проект. Исходники приложений, как и скрипты, доступны на сайте.

Подготовка устройства

Для начала, скачайте в магазине Google Play приложение *Complete Linux Installer*. Либо посетите сайт проекта (http://linuxonandroid.org) и скачайте версию с открытым кодом (но тогда нужно зайти в меню Settings > Security [Настройки > Безопасность] и выбрать Unknown Sources [Неизвестные источники], чтобы разрешить установку приложений не из магазина Play).

Приложение требует наличия прав администратора устройства Android (см. *Руководство по получению root-доступа на Android* на стр. 29). Способ их получения зависит от модели телефона [см. рубрику "Android" **LXF178/179**, стр. 22, для получения более подробных инструкций].

Помимо приложения от проекта LinuxOnAndroid, вам также необходим клиент VNC, чтобы использовать графический рабочий стол. Проектом рекомендуется популярное приложение *Android VNC Viewer*. Кроме того, вам понадобится ещё одно приложение для доступа к встроенной командной строке Android; но если вы обладаете правами root, то оно у вас уже есть. В любом случае, проект предлагает использовать *Android Terminal Emulator*.

Установив их, можете запускать приложение *CompleteLinuxInstaller*. Нажмите на иконку проекта в левом верхнем углу экрана, и откроется меню навигации, состоящее из нескольких пунктов. Важнее всего первые два: Install Guides [Руководство по установке] поможет с настройкой и установкой различных поддерживаемых дистрибутивов Linux, а Launch [Запуск] откроет встроенный загрузчик, через который их можно запустить. Остальные пункты содержат различную информацию и советы вам в помощь.

Чтобы начать процесс установки дистрибутива, запустите *CompleteLinuxInstaller*, откройте меню навигации и выберите Install Guides. Появится список дистрибутивов, доступных для вашего устройства. Выбрав, какой именно вы хотите установить, нажмите на его значок для получения дальнейших инструкций. Допустим, это будет Ubuntu 13.10. Однако процесс будет в целом один и тот же для всех.

Руководство по получению root-доступа на Android

Говоря попросту, наличие прав root (он же — суперпользователь или администратор) даёт вам доступ к той части ОС, куда производитель устройства не хочет, чтобы вы совались. То есть вы получаете больше привилегий и контроль над самыми низкими уровнями Linux-подсистемы Android.

Для этого надо поменять некоторые ключевые компоненты внутри устройства, начиная с загрузчика. По умолчанию загрузчик на телефоне заблокирован и стартует только проверенную систему (нечто вроде функции Secure Boot настольного ПК). Чтобы получить root-доступ, необходимо разблокировать загрузчик, заменить систему восстановления и установить приложение суперпользователя.

Для начала, установите инструмент *android-tools-fastboot*, используя менеджер пакетов вашего дистрибутива. Он позволит взаимодействовать с более глубокими уровнями устройства. Затем устройство нужно перезагрузить в режим быстрой загрузки [Flashboot]. Найдите в Интернете нужную комбинацию клавиш для вашего устройства. Процесс разблокировки загрузчика также варьируется в зависимости от производителя устройства. Для устройств Google, типа Nexus 4, вы можете сделать это с помощью простой команды **fastboot oem unlock**. Для других устройств потребуется получить ключ от производителя.

Затем выбираем другой образ системы восстановления, подходящий для вашего телефона, от популярного проекта ClockWorkMod на www.clockworkmod.com/rommanager. Получив нужный файл IMG, прошиваем его с помощью команды **fastboot flash recovery <image name>**.

Наконец, берём приложение для управления разрешениями суперпользователя, например, *Superuser* от проекта CWM или популярный *SuperSU*.



› **В устройстве Samsung права суперпользователя легко добыть с помощью графического приложения *Heimdall.***

Установка дистрибутива

Когда вы выберете дистрибутив, приложение предоставит подробные инструкции, разбитые на четыре страницы. На Странице 1 описываются основные требования к устройству для запуска дистрибутива Linux. Самое главное здесь — убедиться, что вы включили поддержку отладки. Для этого зайдите в Settings > Applications > Development [Настройки > Приложения > Разработка] и проверьте, что параметр USB Debugging [Отладка по USB] включён.

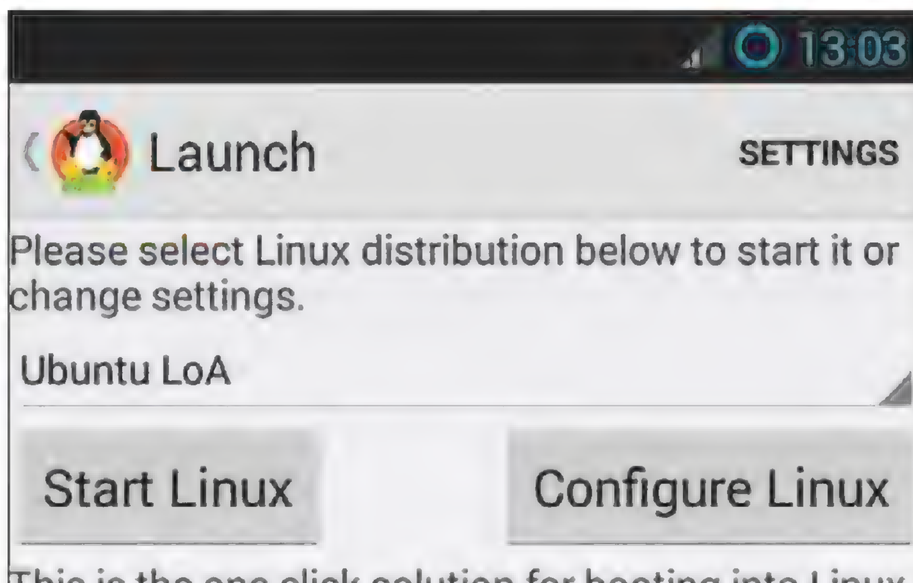
Страница 2 содержит ссылки на загрузку образа выбранного дистрибутива. При нажатии на кнопку Download [Скачать] всплывет окно еще с тремя кнопками, для скачивания одного из трех поддерживаемых типов образов, о которых говорилось выше: Large, Small или Core. Кроме того, страница предоставляет подробную информацию по ним: размер сжатого образа, который вы будете загружать, и сколько места он займёт на вашей SD-карте после распаковки.

Щелкнув на выбранном образе, вы получите ещё одно всплывающее окно с кнопками, для скачивания его с зеркала SourceForge или через торрент. Если вы выберете второе, приложение скачает торрент-файл, который затем нужно открыть через торрент-клиент и скачать сам образ.

Можно также скачать файл на компьютер, а затем перенести его на на карту памяти на вашем устройстве Android. В этом случае спокойно пропустите инструкции на данной странице. Распаковать загруженный файл можно как на компьютере, так и на самом устройстве. Во втором случае вам понадобится универсальный файловый менеджер — например, платное приложение *Root Explorer* или бесплатный *ES File Explorer*.

Хотя сжатые файлы меньше и быстрее скачиваются, помните, что они сильно сжаты, и (в зависимости от характеристик вашего устройства Android) распаковка может оказаться долгой. А если вы переносите образы на устройство вручную, лучше поместить их в папку. Например, скачав файл Ubuntu 13.10, лучше распаковать его в папку с именем **Ubuntu** на SD-карте.

Затем переходите к Странице 3, содержащей советы по распаковке загруженных файлов. Дальнейшая информация касается запуска извлеченных образов. После загрузки вы попадёте в окно терминала. На Странице 4 перечислены инструкции по управлению процессом установки Linux через клиент VNC. Не забудьте про пароль, указанный на этой странице: он понадобится, чтобы подключиться к серверу VNC внутри вашего дистрибутива.



› **Приложение *CompleteLinuxInstaller* превращает установку дистрибутива в пошаговый процесс.**

Скорая помощь

Образы дистрибутива можно скачать из приложения, а можно с сайта проекта — и перенести на телефон.

Загрузка дистрибутива

Теперь всё готово для загрузки нового дистрибутива. Откройте меню навигации, нажав на иконку приложения или проведя по экрану слева направо и нажав на пункт Launch [Запуск]. Откроется программа-загрузчик дистрибутивов. Для выбора нужного вам воспользуйтесь выпадающим меню на странице.

Если вы получаете сообщение об ошибке, что такого образа не существует, значит, приложение не в состоянии найти извлеченные файлы образа для данного дистрибутива. Возможно, вы не распаковали файл или где-нибудь не в том месте его сохранили. Тогда вам нужно будет указать на файл IMG вручную. Нажмите Settings в верхнем правом углу внутри экрана Launch; »

появится выпадающее меню. Нажмите кнопку Add [Добавить], чтобы открыть страницу для добавления вашего дистрибутива. Введите имя дистрибутива в соответствующем поле и нажмите на кнопку с тремя точками, чтобы найти нужный файл IMG в файловой системе вашего Android и указать на него. По окончании нажмите Save Changes [Сохранить изменения].

Теперь ваш дистрибутив также появится в выпадающем меню на странице Launch. Выбрав его, вы увидите кнопку запуска. Нажмите на неё, чтобы загрузить дистрибутив. Откроется терминал

› Для загрузки Linux установщику понадобится *BusyBox*, которую он установит при первом запуске.

от имени root. Нажмите Enter на виртуальной клавиатуре, чтобы начать процесс загрузки дистрибутива.

Поскольку это первый запуск, вам предложат задать пароль для пользователя по умолчанию (в Ubuntu он по умолчанию получает имя ubuntu). Обратите внимание, что это пароль для учетной записи, а не для входа в сеанс VNC.

Дале вам предложат запустить сервер VNC (для просмотра графического рабочего стола) и сервер SSH (для удаленного доступа к дистрибутиву через защищенное соединение). Дабы максимально использовать преимущества дистрибутива Linux на Android, советуем сделать и то, и другое.

Потом вам предложат задать разрешение экрана для сеанса VNC. Оно может быть любым, но лучше, чтобы оно совпадало с разрешением на устройстве. Однако помните, что при вызове виртуальной клавиатуры часть вашего рабочего стола будет скрыта. Если у вас устройство с большим экраном, обратитесь к инструкциям на wiki проекта, чтобы подстроить размер экрана под размещение программных клавиш.

Вот и всё с настройками. Приложение напомнит вам сохранить их как настройки по умолчанию. Лучше так и сделать, чтобы не отвечать на одни и те же вопросы при каждом запуске дистрибутива. Чуть дальше мы покажем вам, как их можно изменить Сейчас же нужно просто сохранить и позволить приложению завершить загрузку.

По завершении вы увидите запись с необходимыми параметрами для подключения к этому дистрибутиву через VNC и стандартное окно терминала. Если вы хорошо знакомы с интерфейсом командной строки Linux, можете использовать эту оболочку для работы с дистрибутивом, как с любым вариантом дистрибутива на настольном ПК.

В целях безопасности, для начала обновите пароль суперпользователя. Введите в оболочке

```
passwd
```

и новый пароль для суперпользователя. Если вы просили LinuxOnAndroid запускать сервер SSH при загрузке вашего дистрибутива, теперь вы можете подключиться к нему с любого компьютера в сети. Команда

```
ifconfig
```

в терминале на вашем устройстве Android выведет IP-адрес устройства — например, 192.168.2.101.

Чтобы подключиться к устройству, запустите терминал на другом компьютере и введите

```
ssh ubuntu@192.168.2.101
```

Потребуется ввести пароль пользователя ubuntu, который вы назначили при первой загрузке дистрибутива.

Запускаем Linux без прав суперпользователя

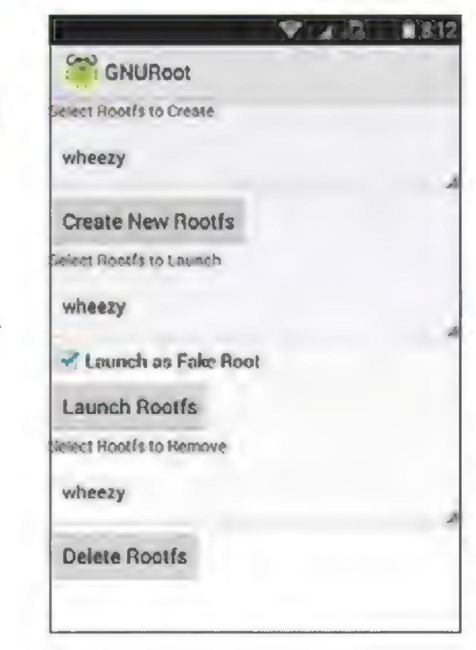
Вы можете запустить Linux на устройстве Android даже без прав суперпользователя, благодаря приложению *GNURoot*. Несмотря на свое название, это приложение не требует таковых. Существенное отличие *GNURoot* от *CompleteLinuxInstaller* в том, что оно дает в дистрибутиве Linux доступ только к интерфейсу командной строки.

Чтобы запустить Linux на вашем Android с помощью *GNURoot*, надо скачать из Play Store его и вспомогательное приложение для конкретного дистрибутива. Таковые существуют для многих популярных дистрибутивов, например, Debian, Gentoo и Fedora. Не забудьте, что в отличие от *GNURoot*, размером всего в несколько КБ, эти приложения весят добрые 100 МБ.

Установив и то, и другое, запустите *GNURoot* и воспользуйтесь выпадающим меню в верхней части страницы для выбора дистрибутива, который вы хотите загрузить.

Убедитесь, что для него скачано вспомогательное приложение. Затем нажмите на кнопку Create New Rootfs [Создать новую Rootfs] для установки дистрибутива; это займет некоторое время, зависящее от ресурсов вашего устройства.

По окончании, нажмите кнопку Launch Rootfs; она откроет терминал, и вы войдёте в систему. Если перед запуском дистрибутива вы выберете флажок Launch as Fake Root [Запуск от имени фиктивного Root], то войдете с привилегиями суперпользователя.



› *GNURoot* допускает использование менеджера пакетов командной строки. Графический интерфейс для рабочего стола пока в разработке.

Для просмотра графического рабочего стола вашего Linux-дистрибутива на Android нажмите на кнопку Home, чтобы свернуть терминал, затем нажмите и запустите приложение VNC (например, *androidVNC*).

Приложение VNC попросит ввести всякие настройки, требуемые для подключения к дистрибутиву. Введите localhost в том поле, где нужно указать IP-адрес сервера VNC, и 5900 в качестве порта. Кроме того, нужен пароль для аутентификации на сервере VNC. Он зависит от дистрибутива. Для образов Ubuntu, пароль ubuntu. Для Arch Linux — это Archlinux, для Debian — Debian, и т. д. Найти пароль можно на wiki-странице дистрибутива на сайте проекта LinuxOnAndroid.

Учтите, что если вы используете *androidVNC*, приложение предложит изменить цветового формата на 24-битный (4bpp). Введя всю информацию, нажмите кнопку Connect [Подключение], чтобы начать сеанс VNC, и вуаля! Клиент VNC подключится к устройству, и вы получите образ графического рабочего стола, который работает внутри запущенного вами дистрибутива.

Опять же, разработчики приложения советуют пользователям *androidVNC* изменить режим ввода на сенсорный. Для этого нажмите на кнопку меню внутри сеанса VNC. Вы увидите кучу вариантов; нужно нажать на режим ввода [Input Mode], а затем выбрать из списка Touchpad.

Теперь вы можете взаимодействовать с рабочим столом, как на ПК. В приложении *androidVNC* однократное нажатие приравнивается к левому щелчку мыши. Для правого щелчка жмите дважды. При нажатии и удерживании приложение отображает три кнопки: две для увеличения и уменьшения, ещё одна — для отображения виртуальной клавиатуры.

> «Можно применять менеджер пакетов для установки приложений.»

Можно также применять свой менеджер пакетов для установки новых приложений. Помните, однако, что дистрибутив сумеет установить только приложения, портированные для архитектуры ARM; но это почти все самые популярные приложения.

По окончании работы завершите сеанс VNC, выйдя из рабочего стола. Затем откройте панель приложений и нажмите на сеанс терминала, выполнявшийся в фоновом режиме. Введите **exit**, чтобы завершить работу дистрибутива Linux. В процессе, вам придется ввести команду **exit** еще пару раз, чтобы выйти и закрыть сеанс терминала.

Расширенные настройки

Вышеописанные шаги позволяют скачать, настроить, загрузить и использовать любой дистрибутив, поддерживаемый на вашем устройстве. Однако приложение предлагает еще и варианты настроить его под себя и сделать удобнее в использовании.

Для начала, можно изменить настройки дистрибутива, указанные при первой загрузке. Чтобы сделать это, выберите дистрибутив, который вы хотите настроить, из выпадающего списка в разделе Launch. Помимо кнопки Start Linux [Запустить Linux], вы увидите кнопку Configure Linux [Настроить Linux]. Нажатие на нее откроет страницу настроек. Здесь вы найдете флажки для включения и отключения серверов SSH и VNC. Также можно изменить разрешение экрана VNC, выбрав из перечисленных вариантов.

На странице есть несколько интересных опций, в том числе флажок Create 1GB SWAP File [Создать swap-файл в 1ГБ]. Как легко догадаться, при его включении приложение создаст файл подкачки для используемого дистрибутива. Но прежде чем включить эту опцию, убедитесь, что у вас на SD-карте хватит места.

Приложение также может монтировать папки из файловой системы Android в Linux. Стукните по кнопке Configure Mounts



› **Убедитесь, что у вашего устройства Android процессор ARMv7 — большинство дистрибутивов поддерживают именно эту архитектуру.**

[Настройка монтирования] для запуска редактора. Выберите Options [Опции] в правом верхнем углу, а затем Add Mount [Добавить]. Откроется страница, где нужно вручную указать расположение папки Android, которую вы хотите перенести (например, **/storage/sdcard0/Downloads**) и путь, куда вы хотите её установить (например, **/home/ubuntu/Downloads**).

Папок можно добавлять сколько угодно. Закончив, нажмите на кнопку Save Changes и выйдите из редактора монтирования.

При загрузке дистрибутива *LinuxOnAndroid* смонтирует перечисленные Android папки в указанных местах.

Рано или поздно пространство внутри образа у вас закончится, в зависимости от того, как вы будете использовать дистрибутив. Тогда придется создать образ большего размера и скопировать все файлы и папки внутри существующего на новый.

Начните с запуска терминала и создания пустого образа нужного размера. Например, команда

```
dd if=/dev/zero of=ubuntuNEW.img bs=1M count=0 seek=4096
```

создаст 4-ГБ образ с именем **ubuntuNEW.img**. Теперь нужно его отформатировать и создать на нём файловую систему. Поможет команда

```
mke2fs -F ubuntuNEW.img
```

Затем скопируйте файл образа, который вы хотите увеличить, из устройства в компьютер — образ назовем **ubuntuOLD.img**. Теперь создайте на рабочем столе две папки (**ubuntuNEW** и **ubuntuOLD**), чтобы смонтировать туда эти образы. Делается это командами

```
sudo mount -o loop ubuntuOLD.img ubuntuOLD/
```

и

```
sudo mount -o loop ubuntuNEW.img ubuntuNEW
```

По завершении, скопируйте все файлы из старого образа на новый. Команда

```
sudo cp -rp ubuntuOLD/* ubuntuNEW
```

позволит скопировать файлы и папки вместе с их правами собственности. Напоследок, просто размонтируйте образы с помощью

```
sudo umount ubuntuOLD
```

и

```
sudo umount ubuntuNEW
```

Теперь вы можете передать файл **ubuntuNEW.img** в телефон, удалить существующий образ и с удовольствием убедиться, что в строке вашего дистрибутива в меню Launch появился новый образ. LXF

Скорая помощь

При удалении дистрибутива проверьте, что он не используется, и затем удалите папку с соответствующим файлом IMG.

Сравнение

» Каждый месяц мы сравниваем тонны программ — а вы можете отдыхать!

Chromebook'и

Генри Винчестер берется за компьютеры, разработанные для неоднозначной операционной системы Chrome.



Про наш тест...

Поскольку по своей природе Chromebook'и основаны на браузере, напрямую установить тестирующие пакеты или проверить производительность 3D невозможно. Однако существует очень милое решение в виде Peacekeeper от 3DMark, сильного сравнительного теста браузеров, который включает HTML5, видео, обработку 3D и тесты игр. У него есть отличное свойство: он может работать непрерывно, истощив батарею и выдав в самом конце результат со средней оценкой. Один Chromebook проходил сравнительный тест 105 раз, так что мы уверены в точности результата.

Тем не менее мы еще запустили сравнительный тест Java SunSpider для проверки на аномалии, а также и тесты реальной работы — например, открытия множества окон браузера. Для оценки производительности мультимедиа были использованы разные форматы файлов, а тест ЖК-монитора от lagom.nl оказался полезен для анализа экрана.

Наша подборка

- Acer C720P
- Asus C200
- HP 14-q005sa
- Samsung XE503C12
- Toshiba CB30-102

Chromebook'и весьма интересны для Сравнения Linux Format, но нам придется пересмотреть свои правила. Большинство обзоров Chromebook'ов сравнивают их с другими ноутбуками, а это не годится для маломощных портативных компьютеров от Google. А вот сравнивая их друг с другом, мы получаем возможность выбрать самые лучшие и самые худшие функции Chromebook'ов, не потерявшись в обширном мире ноутбуков.

Chromebook'и некоторым образом похожи на гоночные машины Формулы-1; Google четко сформулировал, что́ изготовители должны в них вложить, а всякие излишества, чтобы разнообразить их по скорости или внешнему виду — уже дело изготовителей. Все они работают на одной ОС и не имеют 32- и 64-битных вариаций; тестирование жесткого диска тоже отпадает, поскольку у всех он на 16 ГБ, и в трех из наших Chromebook'ов были одинаковые процессоры Intel.

Контраргументом этому будет то, что Chromebook'и — по сути, антипод машин Формулы-1: они дешевые и маломощные и скорее похожи на картинги, с которых начинали свою карьеру гонщики. Но, как и былые нетбуки, эти небольшие машинки не стоит недооценивать, и нас приятно удивило количество функций и порадовали их уверенность и предсказуемость. Если вы списали Chromebook'и со счетов как ограниченные и бесполезные — быть может, наше Сравнение вас переубедит.

> «Chromebook'и: нас порадовали их уверенность и предсказуемость.»

По обработке

Какую мощь они задействуют?

Уровни мощности Chromebook'ов должны восприниматься как данность. Три из всех компьютеров здесь — Acer, Toshiba и HP — работают на одном и том же процессоре Intel Celeron 2955U, настроенном на 1,4 ГГц. Asus повысил этот показатель до 2,42 ГГц, благодаря новому процессору Intel Atom (с кодовым названием Bay Trail), а Samsung предлагает Exynos, индивидуально настраиваемый чип на основе ARM, который работает на 1,9 ГГц. При таких спецификациях мы ожидали, что первые трое выдадут одинаковые результаты, у Asus результаты будут выше, а у Samsung чуть похуже.

Но жизнь решила иначе. Peacekeeper, который провел ряд интенсивных тестов в браузере *Chrome*, сообщает, что процессор Samsung ARM имеет наименьшую мощность и демонстрирует, что технологический гигант не смог извлечь все возможности Intel. Что подтверждает и Java Sunspider: здесь результат оказался второй с конца.

Самое интересное начинается с предпоследнего места: Asus C200. Хотя он работает на самой новой архитектуре Intel — и сам является обновлением чипов Atom, на которых раньше работали нетбуки — и в Peacekeeper, и в Sunspider он показал очень низкие результаты. Создается ощущение, что в приоритетах у Intel сейчас продолжительность работы батарей, но низкое потребление мощности вызвало потерю производительности.

Результат Acer C720P ниже, чем у Toshiba и HP. Поначалу это сбивает с толку, так как они работают на одинаковых чипах, но тут есть простое объяснение: сенсорный экран требует большей мощности обработки для цифровой интерпретации наших прикосновений и ударов, и те же самые тесты на обычном C720 дают более ожидаемые результаты.

Результаты Toshiba и HP в Peacekeeper примерно одинаковы, но в тестах Sunspider Toshiba ведет с солидным отрывом. Странновато, поскольку у HP самый большой объем ОЗУ из всех Chromebook'ов (4 ГБ против 2 ГБ у всех остальных); но единственное очевидное различие здесь — это встроенный модем HP. Его вполне может хватить, чтобы несколько снизить производительность процессора.



› Web требует все больших ресурсов, и Chromebook'ам следует не отставать.

Сравнительные тесты Peacekeeper и Sunspider — синтетические, но они вполне позволяют оценить производительность. Мы также попытались удушить Chromebook'и, открыв в них сразу 12 популярных сайтов, включая видео с YouTube с высоким разрешением, несколько экспериментов с HTML5, игру и web-плейер Spotify. В полном соответствии с теоретическими тестами, в победители вышли Toshiba и HP: оба умудрились, не дрогнув, обработать 36 одновременно открытых вкладок, тогда как остальные рухнули уже на 24.

Вердикт

Toshiba CB30-102 — 5 из 5
HP 14-q005sa — 4 из 5
Acer C720P — 3 из 5
Asus C200 — 3 из 5
Samsung XE503C12 — 2 из 5

» Toshiba — настоящий Геркулес, а за ним идет HP.

Подключаемость

Сколько всякой периферии ладит с этой машинкой?

Чтобы не удорожать машинку, Google свел устройства вывода Chromebook'ов к минимуму. Во всех Chromebook'ах есть выход HDMI, в большинстве есть, как минимум, два порта USB — один 2.0 и один 3.0, только у Toshiba нет 2.0, зато есть два 3.0. Похоже, Google пытается вынудить вас отказаться от хранения данных на USB в пользу своей облачной экосистемы, предлагающей бесплатно два года облачного хранения на 100 ГБ. Исключение составляет HP, где один 2.0 и два USB 3.0.

Слоты для SD-карты предусмотрены у всех, кроме Samsung, выбравшего невероятно крошечный формат microSD. На самом деле, это хорошая идея: вы можете расширить хранилище, вставив крошечную карту и забыв о ней, и слот снабжен крышкой, так что карту нельзя случайно выронить.

У всех ноутбуков есть разъем для кенсингтонского замка, чтобы можно было прикрепить ваш Chromebook к столбу, пока вы расхаживаете по супермаркету. Нам это нравится, потому что на ноутбуках такое встречается все реже и реже, а это важно в школьных «джунглях».

Ни в одном Chromebook'е в этом тесте не было порта для зарядки micro-USB, как в HP Chromebook 11; они предоставляют вам самим отыскать проприетарный коннектор, разнобой которых отнюдь не радует.



› Выбор коннекторов на всех Chromebook'ах небогат, и отсутствуют такие порты, как Thunderbolt или VGA.



HP 14-q005sa — 5 из 5
Samsung XE503C12 — 4 из 5
Acer C720P — 3 из 5
Asus C200 — 3 из 5
Toshiba CB30-102 — 3 из 5

» В конечном итоге нас уломал дополнительный USB-порт на HP.

Структура и дизайн

Хорошо ли сделаны, удобны ли эти дешевые небольшие ноутбуки?

Есть некая ирония в присутствии в названии Chromebook'а слова «Хром», коли в основном их делают из пластика. И хотя может создаться впечатление, что из-за этого они дешевые и непрочные, мы обнаружили, что большинство из них весьма прочны и крепко сбиты. Именно такими они и должны быть — ведь дешевизна делает их идеальными для применения в загрязненной рабочей среде или как инструмент выполнения домашнего задания замурзанными детскими ручками.

Кроме того, они отличаются индивидуальностью и характерным внешним видом, поскольку каждый производитель старался внести что-то свое. Некоторые даже предлагают альтернативы цветовой гаммы, что, несомненно, порадует модных подростков, тогда как более сдержанные темные оттенки найдут себе место на офисных столах. Даже если вы планируете установить дистрибутив Linux, важно знать, что ваш ноутбук хорошо сделан и им приятно пользоваться.

Acer C720P



Все Chromebook'и в нашем Сравнении сделаны из пластика, однако именно Acer размером 11,6 дюймов лучше всех имитирует металл. Его «гранитно-серый» оттенок издали выглядит, как оружейный металл, и то, что он вообще-то пластмассовый, выясняется только если до него дотронуться. Он исключительно прочен и основателен, и то, что при весе 1,36 кг он самый тяжелый из всех 11,6-дюймовых Chromebook'ов, лишь добавляет ему солидности.

Нам понравились клавиатура и трекпад, хотя клавиши могли бы быть более податливыми. Вентиляционное отверстие сзади позволяет охлаждать его во время самых интенсивных web-сессий, и когда включается вентилятор, он практически не шумит. Поскольку у этой версии стеклянный сенсорный экран, по его внешнему краю проходит защитная резиновая кайма, для защиты от ударов.





Asus C200



Определенно, самый приятный опыт набора текста на Chromebook: клавиши в точности отвечают силе нажатия, не производя шума, и это самый большой трекпад, который не спотыкается, и самый ровный и гладкий трекпад вне ноутбуков Apple. Матовое покрытие на нижней стороне не позволит ноутбуку скользить — ну разве что на замасленных коленях; зато на черной крышке остается больше отпечатков пальцев, чем в центральном городском отделении полиции накануне Нового года.

При весе 1,13 кг и размере 11,6 дюймов, он достаточно легкий и маленький, чтобы поместиться в рюкзаке, заметно его не отяготив, и очень симпатичный и тонкий, а притом и прочный. Вентиляционные отверстия спрятаны за петлями, и благодаря этому незначительный шум, который он производит, почти не слышен.

Беспроводная связь

Как насчет соединения с глобальными источниками данных?

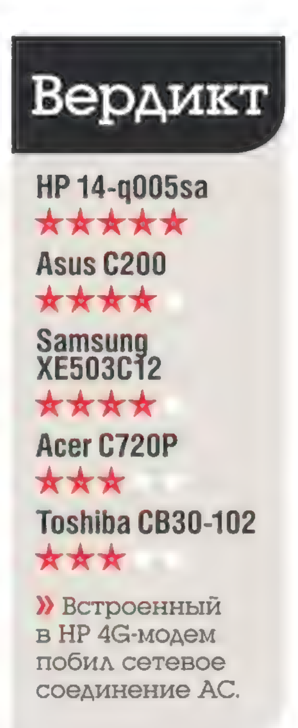
Хотя оффлайн-возможности Chromebook'ов недавно были улучшены, связь с Сетью по-прежнему остается достаточно важной, пусть даже эта связь ограничена возможностью выйти в Интернет для получения почты и скачивания мультимедиа с целью оффлайн-использования.

В свете этого, довольно-таки странно, что ноутбук HP — единственный, который предусмотрел в качестве опционального дополнения 4G-модем. Для остальных придется вне зоны действия Wi-Fi использовать в качестве модема мобильный телефон или обзаводиться устройствами USB. Об этом стоит помнить, если вы проводите много времени в дороге.

Ни у одного из Chromebook'ов не имеется порта Ethernet, но Asus и Samsung включают более быстрое беспроводное сетевое соединение AC.

Не то чтобы протестировать, а даже увидеть скорость вашего соединения нет никакой возможности, что выглядит явным упущением со стороны Google, но мы соединили оба Chromebook'а с нашим модемом AC, и они немного ожили. Хотя скорость оказалась сравнительно приличной, мы обнаружили, что нашему соединению AC не хватает дальности, так что для просторных домов оно навряд ли подойдет.

Все Chromebook'и поддерживают также Bluetooth 4.0, что довольно удобно для передачи файлов с мобильного телефона и на него и беспроводного соединения устройств с колонками и прочими периферийными устройствами.



Вердикт

HP 14-q005sa

Asus C200

Samsung XE503C12

Acer C720P

Toshiba CB30-102

» Встроенный в HP 4G-модем побил сетевое соединение AC.

HP q005sa ★★★

HP недавно выпустил очень симпатичную линию, сумев создать собственную эстетику вне принципов дизайна некой ИТ-компании из Купертино. И их Chromebook не является исключением: наш экземпляр был смелого бирюзового оттенка, и весьма впечатляет, как мягкая на ощупь краска нанесена на абсолютно все участки поверхности, кроме клавиатуры (и экрана, конечно). Кроме того, есть персиковый цвет и старый добрый белый.

При всех своих прорывах в дизайне, HP по-прежнему не удается в полной мере обрести статус производителя бюджетных ноутбуков: у него очень неотзывчивые клавиатура и трекпад. Вдобавок это самый шумный ноутбук в наших тестах, что удивительно, поскольку мы полагали, что 14-дюймовые шасси могут дать больше места для бо́льших по размеру радиаторов и более тихих вентиляторов.



Samsung XE503C12 ★★★★



Крупнейшая в мире технологическая компания вызвала горячие споры, решив приделать к шасси жуткую искусственную кожу. Этот 11,6-дюймовый Chromebook явно пострадал от подобного решения, но на самом деле, все не так уж плохо. Фактически, когда вы чувствуете эту прорезиненную поверхность, создается ощущение прикосновения к дорогому бумажнику, хотя сверхэстетов внешний вид в восторг не приведет.

В общем и целом, он напоминает о 1980-х, с квадратными клавишами в стиле Spectrum, громыхающими при печати, и выполнен в черном цвете, что прекрасно подошло бы элитному программисту, но как-то не очень вяжется с ноутбуком, разработанным в первую очередь для сидения в Facebook. Однако выглядит он прочным, и Samsung перенес на него свой профиль с более дорогих ноутбуков, придав ему соблазнительную тонкость.



Toshiba CB30-102 ★★★★



Несмотря на его серебряное покрытие, возвращающее в прошлое десятилетие, в Toshiba Chromebook много хорошего. Внешние поверхности покрыты тысячами мелких бугорков в стиле Lego, чтобы он не выскользнул у вас из рук. Печатная поверхность кажется грубоватой, шершавой, но это удобно для расширенных сессий электронной почты. И сама по себе клавиатура отличная, с большими клавишами и просторным трекпадом.

При размере 13,3 дюйма, это второй по величине Chromebook в нашем тесте, но его вес 1,5 кг и прочная сборка очень хороши, и когда вы ставите его на колени, он стоит на них крепко. Кроме того, он совершенно тихий: вентилятор только немного слышно при интенсивной работе. И очень приятный, если привыкнуть к его утилитарному внешнему виду.



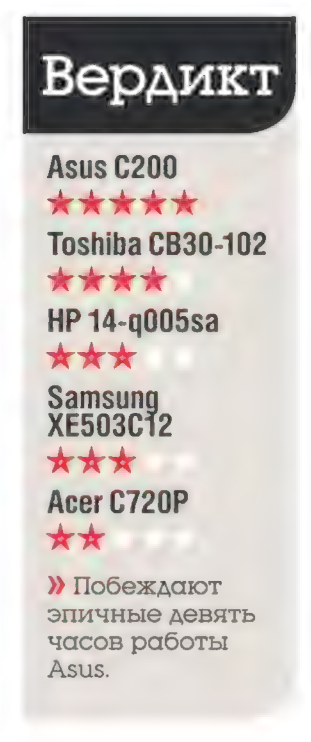
Производительность батарей

Хватает ли батарей Chromebook на заявленный срок?

Маломощные компоненты и основанная на браузере операционная система Chromebook должны означать, что они весьма деликатны в потреблении мощности батарей, растягивая ее на весь рабочий день, а не глотая за пару часов. И это действительно справедливо по большей части для всех Chromebook'ов, которые мы тестировали — даже Acer C720P умудрился проработать пять часов, а это куда больше, чем у большинства виденных нами дешевых ноутбуков. Кроме того, Acer подсчитал, что версия его Chromebook без тачскрина (C720) будет работать на час дольше, что и понятно, с учетом природы постоянного включения стеклянных экранов.

HP Chromebook добавил к результату Acer 40 минут —неожиданно мало, поскольку мы полагали, что его больший размер позволит использовать в шасси более емкие батареи. На самом деле, нам очень интересно, что HP напихал в свой ноутбук — может, свинцовую дробь? Samsung разливался соловьем, расхваливая долгожительство батарей своего процессора Exynos, однако он оказался ровно посередине наших результатов, проработав около шести часов.

Toshiba сумел продвинуться дальше — шесть часов и девятнадцать минут; однако победителем здесь с большим отрывом стал Asus C200, проработав аж девять часов и 43 минуты, при постоянной работе в Интернете. Редко когда продолжительность работы батарей соответствует заявленной производителем; тем не менее, наш результат всего на 17 минут отстал от расчетов Asus.



Вердикт

Asus C200 ★★★★★

Toshiba CB30-102 ★★★★

HP 14-q005sa ★★★

Samsung XE503C12 ★★★

Acer C720P ★★

» Побеждают эпичные девять часов работы Asus.

Качество дисплея

Ласкают ли глаз дисплеи, или на них тошно смотреть?

Экран — та область, на которой производители бюджетных ноутбуков могут прилично сэкономить, и по сравнению с чудесными экранами мониторов у нас в офисе, дисплеи Chromebook'ов в основном ужасны. Однако, поскольку тонким рисованием вы на Chromebook'ах заниматься вряд ли будете, это не очень существенно, и они явно на шаг впереди по сравнению с вызывающими мигрень экранами нетбуков.

У всех экранов в нашем тесте было одинаковое разрешение, 1366×768 пикселей, что является самым низким показателем HD. В форм-факторах Asus, Acer и Samsung при 10,6 дюйма это не так уж важно, но на 13,3-дюймовом экране Toshiba пиксели становятся заметнее, а уж при отображении на 14 дюймах HP выглядят огромными и уродливыми.

Кроме того, HP предлагает худшие уровни черного и весьма среднюю насыщенность белого, но его спасает хорошая работа под прямыми лучами солнца. Samsung, безусловно, знает толк в экранах, благодаря своим чудесным смартфонам и планшетам, и у его экрана Chromebook самые лучшие уровни черного. Матовое покрытие означает, что цвета менее определенны, зато делает использование машинки на улице намного удобнее; мы бы определенно посоветовали именно его, если вы намерены работать под ярким солнцем. Везет же вам...

Экран нашей модели Acer оказался из разряда тех сенсорных экранов, на которые неожиданно подсаживаешься. Однако под прямым солнечным светом пользоваться им практически невозможно, так как для емкостного сенсорного экрана требуются дополнительные слои стекла. Удобство сенсорного экрана и его бесполезность под прямыми солнечными лучами почти взаимоисключающие. И, наконец, экраны Asus и Toshiba заняли место посередине: Toshiba предлагает отличные черные и белые тона, но плохой угол обзора, а у Asus все наоборот.



› На Chromebook вы не найдете всех этих глупостей ultra-HD Retina.

Вердикт

Samsung XE503C12 ★★★★★
Acer C720P ★★★
Asus C200 ★★★
Toshiba CB30-102 ★★★
HP 14-q005sa ★★

» Матовое покрытие обусловило победу Samsung.

Поддержка мультимедиа

Какой из них лучше всего для отдыха?

Наш предыдущий опыт работы с мультимедиа на Chromebook'ах не особо впечатлял: видеофайлы HD воспроизводились с задержками или вообще не воспроизводились. К счастью, Google подсуетился, и все Chromebook'и, которые мы тестировали, воспроизводили разные файлы 1080p MP4.

Чтобы немного усложнить задачу, мы также попробовали погонять видео 1080p 60 кадров в секунду в формате QuickTime, которое теоретически требует в два раза большей мощности обработки. Оно отлично смотрелось на HP, Acer и Toshiba, но совершенно не работало на ноутбуках Samsung и Asus. Оба они очень новые, так что, возможно, проблему решит обновление ПО. Каждый Chromebook снабжен разъемом для наушников и микрофона или гарнитуры смартфона, чтобы слышать или записывать свой голос. Кроме того, в них, конечно, есть колонки, но, как и экран, это область, где можно снизить стоимость.

Лучшим здесь стал HP, предлагающий самый отличный звук — без сомнения, благодаря акустике его большего корпуса. Это не относится к Toshiba: его колонки сильно уступают Asus, который предлагает достойные низкие звуки в ущерб высоким. Колонкам Samsung не хватает объема, а колонки Acer звучат так, словно находятся в поле на расстоянии 30 миль от вас.



› С контентом YouTube на всех наших Chromebook'ах проблем не было.

Вердикт

HP 14-q005sa ★★★★★
Acer C720P ★★★
Asus C200 ★★★
Toshiba CB30-102 ★★★
Samsung XE503C12 ★★

» HP побеждает благодаря хорошей поддержке звука и форматов файлов.

Chromebook'и

Вердикт



› HP Chromebook получает золото — и слава Богу, что он бывает и белого цвета, на случай чрезмерной бирюзовости.

Выяснилось, что идеального Chromebook'а не существует, и добавочные функции, например, тачскрин, влияют на их работу негативно. Неплохим принципом будет избегать излишеств, придерживаясь базовых моделей.

Победителем стал HP Chromebook 14, лишь чуть-чуть опередив остальных. Он успешно прошел наши тесты; его солидные колонки отлично воспроизводят мультимедиа, у него есть все нужные порты; и все это дополнено 4G. Нам нравится его четкий дизайн и смелая цветовая гамма — однако прежде чем его покупать, стоит подумать: не слишком ли она смелая?

Среди 11,6-дюймовых лучшим оказался Asus C200, сочетающий отличное качество сборки с супербыстрым сетевым соединением AC. Слабое место у него одно — мощность процессора, которая, однако, позволяет ему 10 часов работать на батареях; кому нужно нечто способное проработать целый день — вы его нашли. С очень малым отрывом от Asus Chromebook идет Toshiba. Несмотря на суровый дизайн, он отлично прошел наши тесты и превосходно показал себя в мультимедиа, а заодно доказал, что ОЗУ — не самое главное в странном мире Chromebook'ов: имея всего половину от 4 ГБ HP, он все же прошел наш истощающий память тест в браузере. Acer C720P кажется достойным, но слегка тормозится своим модным тачскрином: тот быстро тратит батареи и создает лишнюю нагрузку на процессор. Тачскрины не очень нужны в ноутбуках (только не говорите об этом Microsoft), так что мы бы советовали стандартный C720, если вам нужна мощность в небольшой упаковке.

И мы надеемся, что за формат Chromebook'а примется Samsung, хотя решение взять свой процессор ARM стало катастрофой. ARM'ы годны в более настраиваемой и индивидуальной среде смартфонов, но против других Chromebook'ов слабы и толком не обеспечивают ни производительности, ни выживаемости батарей.

> «Отлично для воспроизведения мультимедиа, есть нужные порты, и дополнено 4G.»



I HP 14-q005sa ★★★★★

Сайт: www.hp.com Цена: £300

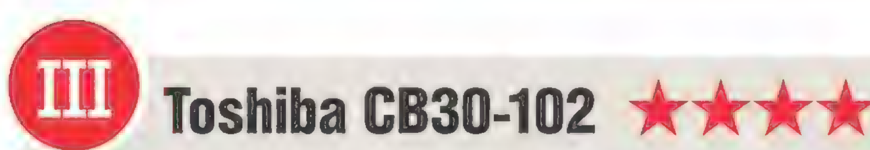
» Chromebook, оправдывающий свой смелый дизайн огромной мощностью.



II Asus C200 ★★★★☆

Сайт: www.asus.com Цена: £200

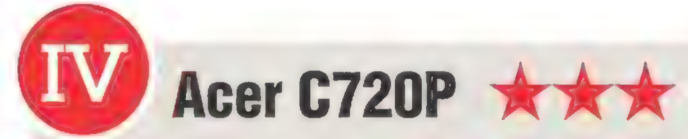
» Невероятная длительность работы смягчает посредственную скорость.



III Toshiba CB30-102 ★★★★☆

Сайт: www.toshiba.co.uk Цена: £230

» Солидный, прочный Chromebook с достойной мощностью.



IV Acer C720P ★★★☆☆

Сайт: www.acer.co.uk Цена: £250

» В нем много хорошего, но все это портит тачскрин.



V Samsung XE503C12 ★★☆☆☆

Сайт: www.samsung.com Цена: £250

» Индивидуальный процессор стал для этого Chromebook роковым.

Обратная связь

Что вы думаете? Каким был ваш опыт работы с Chromebook — хорошим или плохим? Напишите нам на lxf.letters@futurenet.co.uk.

Рассмотрите также...

Мы не включили в наше Сравнение Dell Chromebook 11, поскольку на момент написания они были невероятно редки. Dell позиционировал их, в том числе и по цене, как устройства для обучения, но, когда они появятся в массовом количестве, мы полагаем, что они подойдут не только детям. Dell имеет солидный дизайн, и выбрав версию 4 ГБ, вы получите отличную производительность от двуядерного процессора Intel Celeron с возможностью работы батарей до восьми часов.

Если вас не вполне устраивают предлагаемые на данный момент Chromebook'и, вам, возможно, имеет смысл немного подождать. Google объявил о встраивании поддержки приложений Android для I/O в этом году, и это означает, что в будущем ценной функцией Chromebook'а станет сенсорный экран. Например, Lenovo объявили о выпуске N Series, которая ко времени выхода журнала уже должна появиться в Великобритании. Эта серия включает N20P, с тачскрином и гибким креплением, которое позволяет развернуть дисплей, чтобы удобнее было смотреть Netflix.

Acer тоже обновляет свой популярный C720, который станет C730 (или CB3), с опционным выбором двух- или четырехъядерного процессора Bay Trail. LXF

ВАШИ ПРОБЛЕМЫ С LINUX РЕШЕНЫ!

Могучий интеллект Linux, Нейл Ботвик, рассматривает наиболее общие проблемы и предлагает ряд решений, которые наверняка заставят вашу систему работать без проблем и перебоев. В основном.

В какой-то момент у всех возникали проблемы с Linux. И очень важно, быстро ли эти проблемы решаются. Замечательный компонент открытого кода и компьютерного сообщества — широчайшие возможности получения помощи онлайн в виде блогов, сайтов и форумов. Без сомнения, это имеет огромную важность при определении и решении проблем, и чудесно видеть, с каким воодушевлением сообщество помогает и просто посторонним людям, и новичкам. Однако ничто не заменит изначального знания.

Чтобы вооружить вас знаниями, необходимыми для нормальной работы ваших систем Linux, мы собрали весь наш многолетний опыт ответов на самые разные вопросы от читателей и посетителей форума www.linuxformat.com и профильтровали его, превратив в богатую смесь отборных знаний по Linux, способную помочь решить основные проблемы, с которыми регулярно встречаются пользователи Linux.

> «С учетом сложности системы Linux, поразительно, как она вообще справляется.»

Компьютеры во многом представляют собой шаткую конструкцию из взаимосвязанных стандартов, протоколов, интерфейсов и файлов. С учетом всей сложности современной системы Linux, поразительно, как Linux-ПК вообще умудряется успешно справляться с деликатнейшими задачами запуска, загрузки, включения ядра, загрузки компонентов файловой системы, прохождения по init и загрузки графического интерфейса X с интерактивным графическим менеджером рабочего стола, беспроводного Интернет-соединения и прочих сервисов, имеющимися в современных системах.

На любом этапе эта зыбкая структура может рухнуть — и если такое произойдет, мы будем рядом, чтобы помочь вам соединить обломки! Наше руководство начнется с самого начала, от решения проблем с загрузкой, оборудованием и системой, затем перейдет на более общие проблемы с внешним оборудованием, сетями и, наконец, на классические проблемы рабочего стола. Мы надеемся, что эти бесценные крупицы знания Linux развеют для вас окружающий мрак и помогут сделать вашу работу в Linux хотя бы немного лучше и даже приятнее. Итак, перевернем страницу и приступим к решению ваших проблем! »

Приступаем...


Ваши проблемы с Linux решены!
Нужна помощь?
Вы найдете дружеский совет на linuxformat.com /forums



Приступим

Проблемы при загрузке? Рассмотрим самые общие.

Остановка при загрузке

В Мой компьютер не грузится. Появляется экран-заставка, и на этом все останавливается.

О Для начала — проявите терпение: возможно, процесс загрузки пытается инициировать часть оборудования или получить доступ к сети до того, как она стала доступной. Это может прекратиться через пару минут, и тогда либо продолжится процесс загрузки, либо вы получите сообщение об ошибке, с которым сможете работать. В старые добрые времена дистрибутивы при загрузке заполняли экран текстовыми сообщениями, а потом вошло в моду прятать их за экраном-заставкой. К сожалению, заставка прячет также и все сообщения об ошибках. Вы можете отключить экран-заставку в меню загрузки: если ваш компьютер начинает загружаться, не показывая меню, удерживайте Esc, тогда меню вызовется. Если там есть опция восстановления или безопасного режима, выберите ее, поскольку она отключает экран-заставку и часто выбирает более безопасные (хотя потенциально менее эффективные) опции загрузки, и нажмите E для редактирования. Найдите строку, начинающуюся с **linux**, и ищите такие опции, как quiet, splash или theme. Удалите эти опции, и нажмите F10 для продолжения загрузки.

Это не решит вашей проблемы (если только проблема не в самом экране-заставке), но позволит увидеть, что происходит, и получить либо сообщение об ошибке, либо представление о том, на какой стадии загрузки происходит сбой. После этого вы сможете продолжить выяснение. Если окажется, что проблемы — в распознавании оборудования, попробуйте отсоединить все ненужное оборудование, например, принтеры, сканеры или зенитные установки USB. Обнаружив проблему, вы сможете приступить к ее решению.

Ошибки при загрузке

В При загрузке я вижу такие сообщения об ошибках:

```
Pci Express Device Error. Severity NN. Uncorrected non Fatal.
Pci Bus Error
Type:N Transaction Layer. Flow Control Protocol N:First.Receiver 1
```

Это проблема возникает у меня на новой материнской плате, которая прекрасно работает с Windows.

О То, что оборудование работает с Windows, отнюдь не означает, что у него нет проблем. Довольно распространенным является подход: «С Windows оно работает — установите Windows, и мы исправим все прочие проблемы в обновлении». Во всем, что касается материнских плат, первым делом нужно проверить наличие обновлений BIOS. Даже если материнская плата новая, к тому моменту, когда она пройдет по всему процессу отгрузки и распространения с Дальнего Востока до вашего производителя компьютеров, обычно уже появляется новый BIOS. Обновление BIOS способно решить множество проблем с загрузкой, но если ваша проблема никуда не делась, есть и другие шаги.

Вы можете изменить способ обращения ядра к оборудованию через ряд параметров, передаваемых ядру во время загрузки. В меню загрузки *Grub* (нажмите Esc, если оно не появилось по умолчанию) выделите вашу обычную опцию загрузки и нажмите E (для редактирования). Отредактируйте строку, начинающуюся с kernel, добавив несколько или все эти опции в конце строки, разделяя их пробелами:

```
noapic
```

или

```
noapictimer, acpi=off
```

или

```
acpi=force
irqpoll
pci=nommconf
pci=nomsi
```

Обратите внимание на разницу в написании второго и третьего набора опций: все, что есть общего у acpi и apic — это состав букв имени. Нажмите F10, для загрузки опций, которые вы добавили. Найдя комбинацию, которая подружит ваше ядро с оборудованием, можно сделать изменение постоянным, добавив опции в GRUB_CMDLINE_LINUX_DEFAULT и GRUB_CMDLINE_LINUX в **/etc/default/grub**, сохранив файл и обновив *Grub* с помощью одной из следующих команд. Какая это будет команда, зависит от вашего дистрибутива — если не сработает первая, попробуйте вторую, а потом третью.

```
sudo update-grub2
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
sudo grub2-mkconfig -o /boot/grub/grub.cfg
```

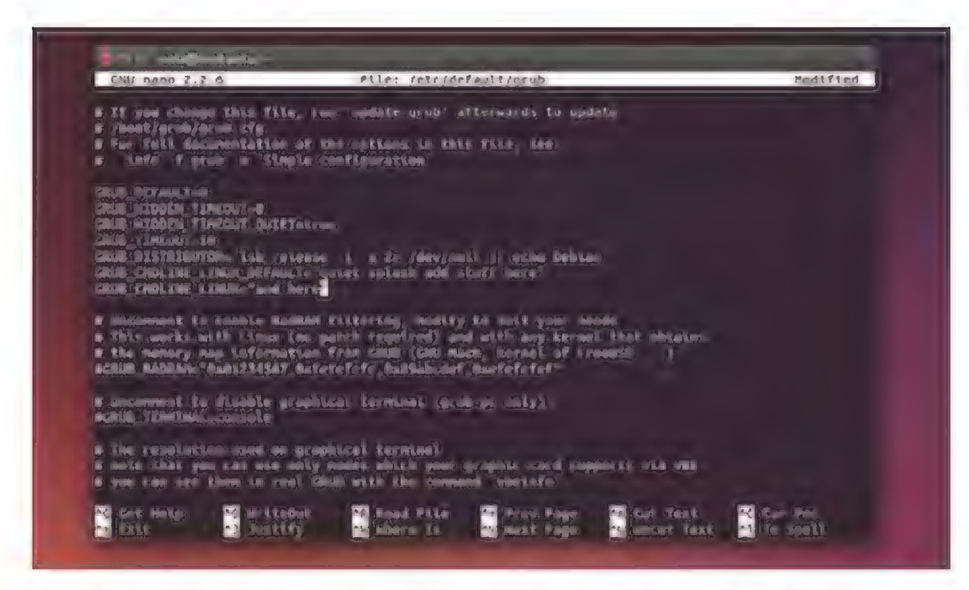
```
[    8.693620] fuse init (API version 7.22)
[info] Loading kernel module ecryptfs.
[info] Loading kernel module fuse.
[....] Setting up LVM Volume Groups...  No volume groups found
done.
[ ok ] Starting remaining crypto disks...done.
[ ok ] Activating lvm and md swap...done.
[....] Checking file systems...fsck from util-linux 2.20.1
done.
[ ok ] Mounting local filesystems...done.
[ ok ] Activating swapfile swap...done.
[ ok ] Cleaning up temporary files....
[ ok ] Skip starting firewall: ufw (not enabled)...done.
[ ok ] Setting kernel variables ...done.
[....] Setting up resolvconf.../etc/resolvconf/update.d/libc: Warning: /etc/
lv.conf is not a symbolic link to /etc/resolvconf/run/resolv.conf
done.
[ ok ] Configuring network interfaces...done.
[ ok ] Starting rpcbind daemon....
[ ok ] Cleaning up temporary files....
[ ok ] Setting up ALSA...done (none loaded).
Running 0dns-down to make sure resolv.conf is ok...done.
[info] Setting console screen modes.
[ ok ] Setting up console font and keymap...done.
```

› Отключив заставку при загрузке вашего дистрибутива, вы сможете посмотреть, что там происходит — или отказывается происходить — при инициализации системы.

Переустановка Windows сломала Linux

В Я установил Linux на свой компьютер с Windows, и он предложил мне выбрать, какую систему использовать при включении. Недавно мне пришлось переустановить Windows, и теперь Linux пропал. Я не хочу переустанавливать Linux, поскольку я потеряю все свои документы и настройки. Есть ли возможность вернуть мне мой Linux?

О Linux для выбора между разными опциями загрузки использует загрузчик *Grub*, однако у Windows — собственный загрузчик. Загрузчик — это небольшой фрагмент кода, встроенный в начало вашего жесткого диска для того, чтобы запустить процесс загрузки вашей операционной системы. При установке Windows установила собственный загрузчик, переписав ваш существующий *Grub*. Не переживайте: пропал только код загрузчика — ваша система Linux и ваши файл по-прежнему на месте, и все, что вам нужно сделать — восстановить *Grub*. Установочные диски некоторых дистрибутивов имеют опцию восстановления



Чтобы внести постоянные изменения в меню Grub, отредактируйте /etc/default/grub и запустите *grub2-mkconfig* или *update-grub*.

загрузчика, или есть более общая опция в Rescatux (на **LXFDVD** этого месяца). Загрузитесь с CD Rescatux и выберите опцию Restore Grub. Она произведет поиск по вашим разделам Linux и предложит вам выбрать раздел, содержащий файловую систему root вашего Linux, а затем спросит, на какой жесткий диск вы хотите установить *Grub* — обычно это **/dev/sda**, первый диск. Некоторое время, способное показаться бесконечным, экран останется пустым, но не поддавайтесь панике и не перегружайтесь: подождите, пока вам скажут, что *Grub* был переустановлен перед перезагрузкой, и после этого должно появиться ваше прежнее меню с доступом к вашей нетронутой системе Linux.

Загрузка нескольких дистрибутивов

В Я знаю, что бывает двойная загрузка с Linux и Windows, но могу ли я сделать то же самое с несколькими дистрибутивами Linux? Если да, то как мне делить между ними свои данные? Как разбить мой диск на разделы?

О Да, это возможно. Раздел root каждому дистрибутиву нужен свой, но вы можете делить между ними всеми один и тот же раздел подкачки [swap]. Рекомендуется наличие отдельного раздела для **/home** — вы вряд ли захотите привязывать все свои данные к одному дистрибутиву, который в будущем, возможно, покинете. Хотя желателен отдельный раздел **/home**, не используйте одну и ту же директорию **home** для всех дистрибутивов, поскольку возможна несовместимость между разными версиями используемых программ. Я предпочитаю добавлять к имени пользователя имя дистрибутива: например, fred-ubuntu, fred-debian, и т. д. Поскольку каждый пользователь имеет тот же самый UID — большинство программ установки делают первого пользователя 1000 — у вас будет к ним доступ с правом записи, и затем вы сможете создать символические ссылки на важные директории данных, например, документы, музыку, фото и т. д., чтобы дать доступ к этим файлам всем дистрибутивам, сохраняя при этом файлы настройки отдельно.

Что касается меню загрузки, выберите один дистрибутив за основной, и пусть загрузкой занимается он. При каждой установке нового дистрибутива пропускайте опцию установки загрузчика, если это возможно. Затем перезагрузитесь в основной дистрибутив и запустите

```
sudo grub-mkconfig -o /boot/grub/grub.cfg
```

для обновления меню загрузки.

Удаление дистрибутива все поломало

В Я удалил разделы Linux с моего компьютера, готовясь установить что-то другое. Теперь я не могу его загрузить. Все, что у меня есть — это сообщение, что *Grub* не может найти файлы, и подсказка grub rescue>. Я даже не могу загрузиться в Windows.

О Хотя начальный код загрузки для *Grub* хранится в Master Boot Record (MBR) в начале вашего жесткого диска, это очень небольшое пространство. Поэтому начальный код *Grub* загружает файлы и информацию по настройке из директории **grub** или **grub2** на разделе загрузки Linux. Когда вы удалили этот раздел, *Grub* стало некуда обращаться. Если вы планируете установить на освободившееся место другой дистрибутив, просто запустите программу установки, и она настроит для вас *Grub*, распознав также и вашу систему Windows, и после перезагрузки все будет хорошо.

Если вы используете место для существующего раздела Windows или Linux, нужно будет настроить *Grub* для другой системы Linux или переустановить загрузчик Windows. Самый простой способ перенастроить *Grub* — использовать опцию Restore Grub на Rescatux live CD; выберите установку Linux, которой вы хотите поручить работу с файлами *Grub*. Восстановление загрузчика Windows включает загрузку диска Windows (или его раздела восстановления) и запуск одной из этих команд в зависимости от версии Windows. Для Windows XP это

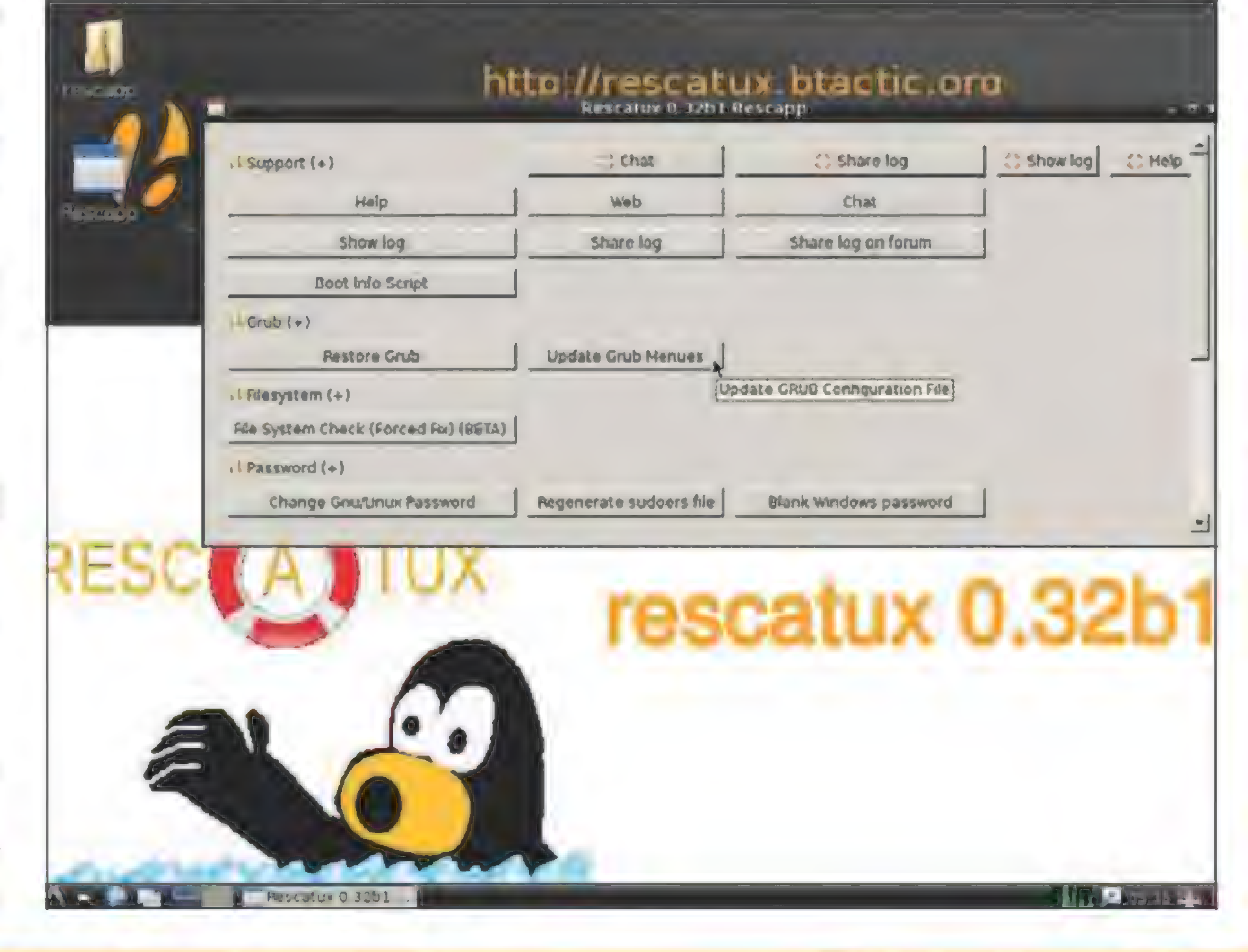
```
fixboot
fixmbr
```

Для Vista и Windows 7 и 8 —

```
bootrec /fixboot
bootrec /fixmbr
bootrec /rebuildbcd
```

> «Желателен отдельный раздел /home, но не используйте один для всех дистрибутивов.»

Если ваш компьютер вообще не загружается, live CD — просто дар божий, так что стоит всегда держать его под рукой. Live CD — это полноценная среда Linux, которая загружается с CD и не требует установки или даже доступа к жесткому диску. Помимо того, что это очень удобно в качестве портативного дистрибутива или экспериментальной среды для знакомства с новыми дистрибутивами без установки, live CD идеален для аварийного восстановления системы. Есть специальные CD аварийного восстановления системы, такие, как System Rescue CD, Rescatux (оба имеются на **LXFDVD** этого месяца) и Grml, но большинство установочных дисков дистрибутивов в наши дни загружаются в среду live, которая содержит достаточно программ, чтобы решить многие проблемы.



Rescatux live CD снабжен удобной кнопкой Restore Grub, благодаря которой вам не придется запоминать редко используемые команды оболочки.

Сложности с «железом»

Укрощение застроптивившегося оборудования.

Поиск драйверов оборудования

В Где найти драйверы для моего оборудования? На сопровождающем диске есть драйверы только для Windows.

О Большей части оборудования не нужны специальные драйверы для Linux: ядро включает драйверы для самых разных устройств. В девяти из десяти случаев ядро распознает оборудование, когда вы его подключите, и загрузит нужный драйвер без вашей помощи. Но у некоторых устройств имеются отдельные драйверы, часто по причине того, что они используют проприетарный код, который нельзя включить в ядро. В подобных случаях live CD, например, Knoppix или System Rescue CD, нередко могут распознать устройство — они включают много дополнительных драйверов. Если устройство является внутренним, обычно оно появляется в данных от **lspci**, а добавление опции **-k** покажет, какой используется драйвер.

Выяснив это, вы сможете найти пакет, где имеется данный драйвер — или с помощью своего менеджера пакетов, или через поиск по Сети. Если он находится в стандартном пакете, не устанавливаемом по умолчанию, что является обычным делом для многих беспроводных драйверов, то все, что вам понадобится для того, чтобы заставить оборудование работать — это установить пакет. Процесс совершенно тот же и для устройств USB, лишь с тем исключением, что вы используете команду **lsusb**, и здесь нет удобного эквивалента параметру **-k**. Обратите внимание, что в некоторых системах запустить ее можно только от имени root.

Если ваше ядро или менеджер пакетов не поддерживают драйверов, вам придется ступить на зыбкую почву скачивания и установки драйверов непосредственно с сайта производителя. Если у них есть подходящие для вашей системы Deb- или RPM-пакеты, то все не так уж плохо. В ином случае найдите соответствующий вашей архитектуре драйвер и следуйте инструкциям по установке. Драйверы должны иметь ссылку на ядро, так что вам, вероятно, нужно будет установить исходник ядра и пакеты *build-essential*.

› Хотя в большинстве дистрибутивов имеется GUI для управления принтерами, пользовательский интерфейс CUPS универсален и хорошо работает.

Периодические сбои

В Периодически у моего компьютера происходят сбои и блокировки — не похоже, чтобы они были связаны с какой-то определенной программой.

О Эта проблема вполне может быть связана с оборудованием, и у нее может быть множество причин. Одной из них может быть бракованная память, а поскольку Linux использует всю имеющуюся память, это более вероятно в Linux, чем в Windows. Загрузитесь с **LXFDVD** и выберите в меню *Memtest*. Пусть он выполнится несколько раз, желательно даже запустить его на всю ночь.

Другой распространенной причиной может быть перегрев: попробуйте запустить мониторинг системы или апплет Taskbar, который показывает температуру в системе. Проверьте, что у вас не засорились вентиляционные отверстия; а если не боитесь взять в руки отвертку, позаботьтесь, чтобы вентилятор и радиатор вашего CPU не были забиты грязью.

Третьей причиной сбоев настольных ПК может стать некачественный блок питания. Дешевые устройства часто со временем портятся, что и приводит к подобным проблемам, а заодно и к повреждению компонентов. Единственным надежным тестом будет просто попробовать использовать другой блок питания.

Проблемы с принтерами

В Мой дистрибутив не распознал мой принтер. Как мне установить драйверы?

О Как вам установить драйверы — и даже есть ли эти драйверы вообще — зависит от вашего принтера. В первую очередь надо заглянуть в базу данных на http://linuxprinting.org. Сделав это до покупки принтера, вы избавитесь от множества проблем. Введите производителя и модель принтера, и база данных

Зачем нужен терминал?

Большая часть советов, которые мы вам даем, предлагается в виде команд терминала. Хотя в большинстве дистрибутивов имеются собственные программы настройки, они значительно различаются, тогда как основные команды одинаковы во всех дистрибутивах. Среднестатистический пользователь смело может запускать эти команды напрямую: они подойдут для всех дистрибутивов, и сработают, даже если ваша проблема не позволяет запуститься рабочему столу.

Некоторые команды надо запускать от имени root, и это можно сделать, либо добавив к команде префикс **sudo** (когда вам предложат ввести свой пароль), либо запустив сначала в терминале **su**, после чего у вас спросят пароль root. Мы будем использовать в этой статье **sudo**, потому что в некоторых дистрибутивах имеется только эта опция. Если у вас есть полный доступ root через **su**, просто запустите команду без **sudo**. Команды, данные без **sudo**, должны запускаться от имени обычного пользователя.

сообщит вам, поддерживается ли он и какие драйверы использовать. CUPS, система печати, используемая в Linux, снабжена множеством драйверов, так что во многих случаях вам не придется ничего устанавливать. Еще больше драйверов в пакете *gutenprint* — http://linuxprinting.org сообщит вам, нужны ли они вам. HP предоставляет драйверы с открытым кодом, так что если у вас модель HP, установите пакет *hplip*, чтобы получить доступ ко всем драйверам для принтеров и сканеров HP. Другие производители предоставляют двоичные пакеты с закрытым кодом. Выясните, как ваш дистрибутив предлагает их — в виде пакета или в ином виде; в качестве крайнего средства вам придется скачать и установить их в соответствии с инструкциями на сайте производителя принтера.

Установив должные драйверы, вы сможете настроить принтер либо с помощью инструмента администрирования принтеров вашего дистрибутива, либо через web-интерфейс CUPS. Чтобы использовать последний — это будет работать в любом дистрибутиве — введите в своем браузере https://localhost:631, войдите в систему, если потребуется, щелкните по вкладке Administration, затем Add Printer, и следуйте инструкциям, не забывая об уже полученной информации о том, какой драйвер использовать лучше всего.

Опция твердотельных дисков

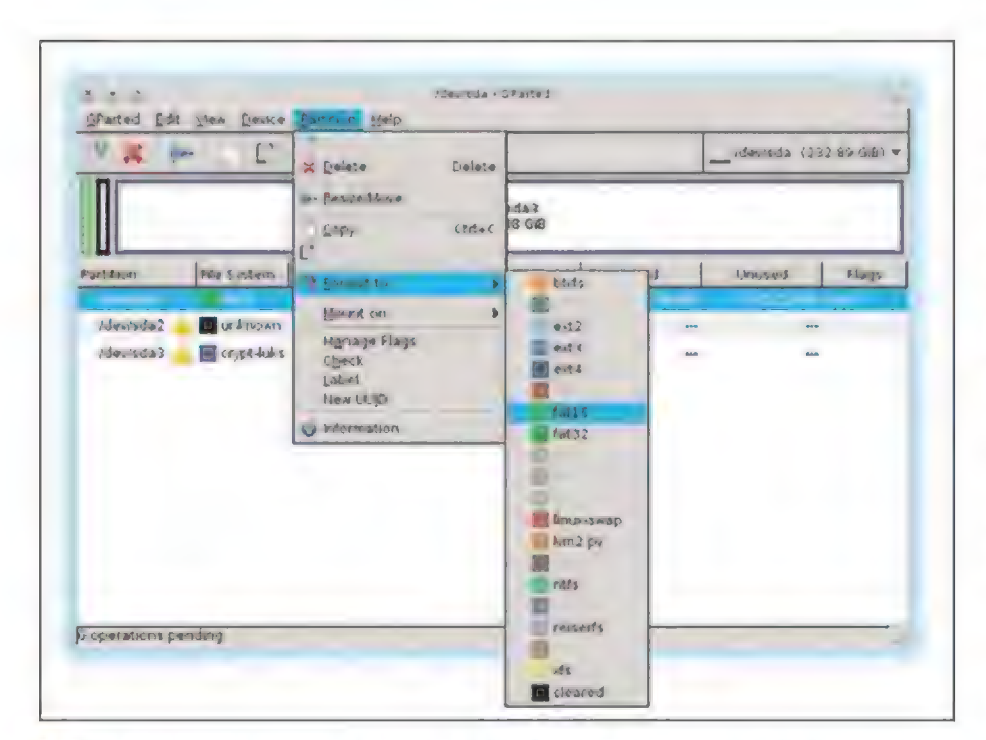
В Мне бы хотелось заменить свой жесткий диск на SSD. Как мне его настроить, и есть ли моменты, где надо проявлять особую осторожность? Я думаю о проблемах, вызванных слишком частой записью во флэш-память. Как насчет TRIM?

О В настройке SSD и вращающегося диска особой разницы нет. В наши дни инструменты разбиения диска на разделы по умолчанию используют параметры, безопасные для дисков обоих типов. По статистике сегодня считается, что SSD способен пережить вращающийся диск. Однако не лишним будет добавить в ваши опции монтирования **noatime** в **/etc/fstab**, если вы этого еще не сделали: это позволит также увеличить скорость.

У вас есть несколько опций копирования данных. Вы можете сделать копию своей директории **home** и сделать свежую установку на SSD, или можете просто все скопировать. Для этого вам стоит загрузиться с live CD и использовать SATA с USB-адаптером, если у вас нет лишнего порта SATA. Скопируйте все разделы, кроме **swap**, на оба диска, и используйте *rsync*, чтобы скопировать все с каждого раздела:

```
rsync -a /mnt/old_partition/ /mnt/new_partition/
```

Вам также надо будет переустановить *Grub* в качестве загрузчика, загрузив что-нибудь вроде Rescatux при наличии в компьютере только нового диска. TRIM, очистка неиспользуемых областей хранения, может быть сделана двумя способами. Самый простой — добавить **relatime,discard** в опции монтирования в **/etc/fstab**.



Для удаления данных с жесткого диска без шансов восстановить, отличная опция — *Darik's Boot and Nuke* (*DBAN*).

```
Darik's Boot and Nuke

Warning: This software irrecoverably destroys data.

This software is provided without any warranty; without even the implied
warranty of merchantability or fitness for a particular purpose. In no event
shall the software authors or contributors be liable for any damages arising
from the use of this software. This software is provided "as is".

http://www.dban.org/

* Press the F2 key to learn about DBAN.
* Press the F3 key for a list of quick commands.
* Press the F4 key to read the RAID disclaimer.
* Press the ENTER key to start DBAN in interactive mode.
* Enter autonuke at this prompt to start DBAN in automatic mode.

boot:
```

Разбить на разделы новый SSD, или жесткий диск любого типа, весьма просто с помощью *GParted* — большинство дистрибутивов и live CD его включают.

Однако единственным недостатком будет то, что при выполнении операции trim работа диска замедляется, поэтому многие предпочитают запускать *fstrim*, когда нагрузка на компьютер невелика. Задания *Cron*, раз в неделю запускающего **fstrim -all**, будет вполне достаточно в большинстве случаев. Опция **--all** очищает все файловые системы, которые ее поддерживают, и тихо игнорирует все остальные. Добавьте опцию **-verbose**, и *Cron* будет сообщать вам обо всем, что сделано.

Удаление файлов всерьез

В Я хочу убедиться, что мой диск на самом деле свободен ото всех данных перед тем, как от него избавиться. Я понимаю, что простого удаления файлов недостаточно, но что мне сделать, чтобы бесследно удалить все мои файлы с диска?

О Это зависит от того, как вы намерены избавиться от диска. Если вы хотите его выбросить, то удар тяжелым молотком точно будет летальным для ваших данных; однако есть и более утонченные — хотя и не столь надежные — методы безопасной очистки диска, которые позволят впоследствии его продать.

> «По статистике сегодня считается, что SSD способен пережить вращающийся диск.»

Вы правы, полагая, что удаления файлов недостаточно, потому что оно просто удаляет их записи из таблицы директории, но сами данные по-прежнему остаются. Вам нужно перезаписать все данные на диске. Житейская мудрость гласит, что сделать это нужно неоднократно, перезаписывая их разными способами или случайными данными, чтобы противостоять самым решительным попыткам восстановления данных. Для большинства случаев вполне подойдет забивание данных нулями:

```
sudo dd if=/dev/zero of=/dev/sdb bs=4k
```

На системном диске этого не сделать, и если именно его вы хотите очистить, надо запустить команду с live CD. Для душевного спокойствия можете добавить еще и случайных данных, используя **/dev/urandom** вместо **/dev/zero**. Помните, что команда **dd** не спрашивает вас, уверены ли вы, и не проверяет дважды, правильно ли вы указали устройство перед нажатием Enter — здесь вам никто не даст второго шанса.

Если вы по-прежнему желаете пойти по пути неоднократной перезаписи, для успокоения души, попробуйте *Darik's Boot and Nuke* (*DBAN*) live CD с http://www.dban.org; это второй по эффективности инструмент после тяжелого молотка, которым вы заготовили размахнуться.

Проблемы с системой

Решайте типовые проблемы через администрирование.

Диск переполняется

В У меня на жестком диске заканчивается место — такое ощущение, что объем просто исчезает. Куда он девается и что мне сделать, чтобы его вернуть?

О Если у вас более одного раздела, то первым делом надо узнать, который из них страдает, по команде **df**:

```
df -h
```

Она покажет общее, используемое и доступное место на каждом разделе. Если у вас отдельный **/home**, вы тогда узнаете, переполняется ли ваша домашняя директория или системный root. Предпринимаемые шаги во всех случаях одинаковы, но для того, чтобы очистить системную директорию, нужны права root.

Для системных файлов обычными причинами проблем являются **/tmp** и **/var**. Первый используется для вре́менных файлов, но иногда там застревают старые файлы, особенно когда процессы выдают отказ без очистки. **/var/log** содержит системные журналы (лог-файлы) и может переполниться, особенно если какой-то процесс зачастил ошибками. Просмотрите содержание любого большого файла перед тем, как с ним работать. Стоит установить *logrotate*, эта утилита будет контролировать все ваши лог-файлы. По умолчанию она каждую неделю начинает новый лог-файл и сжимает предыдущий, сохраняя записи за последние четыре недели. Это может сэкономить много места.

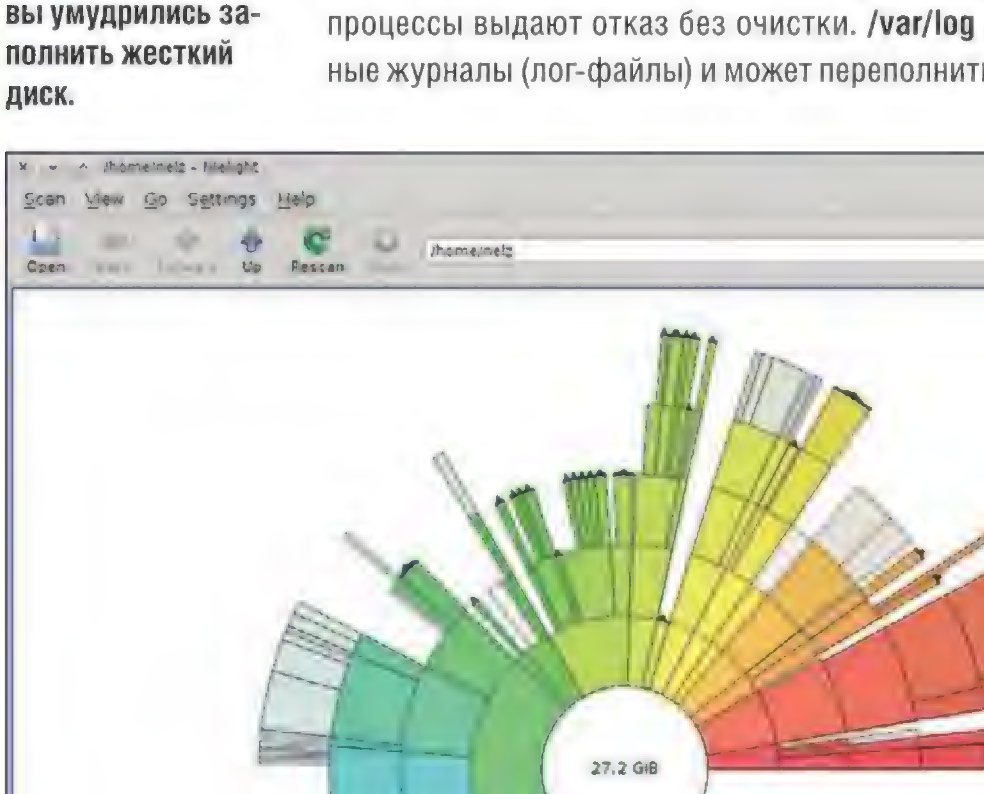
В своей директории **home** проверьте папку **downloads**, кэш браузера и памяти и директорию **.cache**. Можете использовать **du** для проверки объема директорий: команда

```
du -sh * | sort -h
```

перечисляет все в порядке возрастания размера. Желая проверить скрытые директории, скомандуйте

```
du -sh .[0-9a-z]* | sort –h
```

Есть также графические инструменты для визуализации использования диска — *Filelight* для KDE и *Disk Usage Analyser* для Gnome — а *Bleachbit* хорош для выявления и устранения ненужных файлов, например, кэша браузера или других вре́менных файлов.



› *Filelight*, как и *Gnome Disk Usage Analyser*, даст вам неплохую визуализацию, как вы умудрились заполнить жесткий диск.

Куда делась вся моя память?

В Я заметил, что если я немного поработаю в Linux, у меня практически не остается памяти. Куда она девается? Как мне найти причину и что с этим делать?

О На самом деле это нормально: так работает Linux. Свободная память — это память, которая ничего не делает, хотя могла бы помочь производительности вашего компьютера. Ядро Linux распределяет память, которая не используется процессами активно, таким вещам, как кэш файловой системы, чтобы повысить производительность. Вы можете увидеть это, заглянув в данные от **free -h**:

```
total used free shared buffers cached
Mem: 15G 15G 128M 982M 396K 13G
-/+ buffers/cache: 2.3G 13G
```

Как видите, на компьютере с 16 ГБ физической памяти свободны только 128 МБ. Из всей используемой памяти, 13 ГБ используется под буферы. Это значительно ускоряет доступ к диску, благодаря кэшированию записанных данных в памяти и считывания большего объема, чем вам нужно, с учетом вашего следующего считывания. Если эта память понадобится для чего-то более важного, она немедленно будет освобождена.

Ищем неприятности

Linux ведет записи в системном журнале практически обо всем. И если что-то пошло не так, именно в эти записи стоит заглянуть. Записи системного журнала находятся в **/var/log** и обычно называются сообщениями или syslog, в зависимости от программы регистрации, используемой вашим дистрибутивом. Если программа *foo* начала чудить, попробуйте найти её сообщения, командой

```
grep foo /var/log/messages
```

Можете также взглянуть на записи в реальном времени:

```
tail -f /var/log/messages
```

что отобразит новые записи в журнале по мере их появления (**-f** означает “follow” — отслеживать). Запустите в терминале ее, а затем то, что вызывает проблемы, и вы узнаете, что оно говорит. Буфер ядра читается командой **dmesg**, что удобно, если у вас есть проблемы с оборудованием. В ней тоже есть опция “follow”:

```
dmesg --follow
```

удобная при работе с устройствами с «горячим» подключением. Чтобы посмотреть, как определяются встроенные устройства и как они работают на момент загрузки, запустите

```
dmesg | less
```

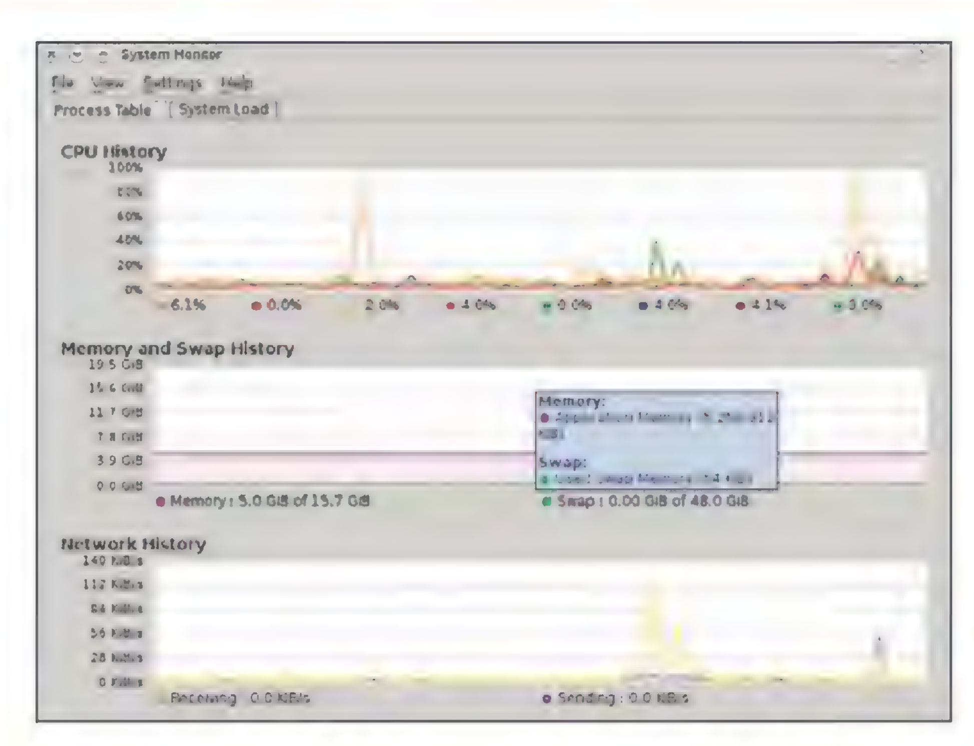
после загрузки, и затем промотайте сообщения или, поскольку их много, нажмите / для поиска соответствующих.

Если ваш дистрибутив использует *systemd*, записи журнала можно отобразить с помощью **journalctl**. Вы можете выполнить ту же самую функцию follow с помощью

```
journalctl --follow
```

или найти ошибки с момента последней загрузки с помощью

```
journalctl -b -p err
```



Используя системные мониторы, убедитесь, что вы смотрите на цифры, не включающие память буферов файловой системы.

В памяти остаются не только данные файловой системы — попробуйте проверить свою свободную память, а затем запустить и остановить какую-либо большую программу, например, *Firefox*, и вы увидите, что количество свободной памяти сократилось. Теперь снова запустите *Firefox* — и вы увидите, насколько быстрее будет второй запуск. Это из-за того, что ядро хранит данные в памяти для следующего использования. То же относится к библиотекам с распределенным доступом, что и является причиной того, что запуск одних и тех же программ из одной среды рабочего стола может быть быстрее, чем запуск из разных сред, особенно если память ограничена.

Переход в home

В **Я слышал, что директорию home лучше держать на отдельном разделе. Почему и как это сделать?**

О Во-первых, обычно собственную файловую систему имеет не ваша индивидуальная директория **home**, а вся директория **/home**, в которой и размещается ваша **home**. На это есть ряд причин, например, упрощение процедуры резервного копирования, но основная причина — хранение ваших данных, как и данных других пользователей, отдельно от операционной системы. Это означает, что вы можете обновить или даже заменить вашу ОС и по-прежнему сохранить все свои индивидуальные настройки и данные.

Самый простой способ сделать это — с помощью *GParted* и отдельного жесткого диска. Жесткий диск должен быть отформатирован в какую-либо файловую систему Linux, чтобы сохранить всю информацию в ваших файлах. Затем используйте *rsync*, чтобы все скопировать.

```
sudo mkdir /media/extdrive/home/
sudo rsync -a --delete /home/ /media/extdrive/
```

Здесь предполагается, что ваш отдельный диск монтирован на **/media/extdrive**. Не пропустите завершающие слэши, это важно. Есть и другие инструменты копирования, но у *rsync* имеется преимущество: он может продолжить работу именно с того момента, на котором был прерван. То есть вы можете продолжать использовать свой ПК в процессе выполнения этой работы, затем загрузиться с live CD и повторить процесс, чтобы обновить файлы, изменившиеся, пока вы использовали систему.

Для следующей стадии нужен live CD, здесь вы будете удалять содержимое **/home** и использовать *GParted* для изменения размера раздела root для достижения оптимальной величины — 20 ГБ должно хватать, однако сделайте его примерно на 50% больше, чем используется в данный момент. Теперь создайте новый раздел после раздела root и отформатируйте его в ext4, если только у вас нет причин выбрать другую файловую систему.

Загрузите файл **/etc/fstab** с раздела root в редактор и добавьте к нему строку

```
/dev/sdaN /home ext4 defaults,noatime 0 0
```

где sdaN — только что созданный вами раздел. И, наконец, скопируйте данные назад с внешнего диска в новую **/home** и перезагрузитесь. Можно сделать все это и без доступа к внешнему диску, но процесс будет несколько сложнее.

Universally Unreadable IDs

В **Меня раздражает использование UUIDs в /etc/fstab. Что случилось со старым методом использования узлов устройств, например, /dev/sda1, и можно ли к нему вернуться?**

О Проблема с узлами устройств в том, что они могут менять имена. Если вы добавите или удалите диск, включая USB-брелки, или измените порядок загрузки BIOS, **sda** может стать **sdb**. UUIDs, с другой стороны, (довольно) уникальные; им неважно, что изменился порядок дисков, эта недружелюбная строковая переменная останется прежней. Тут-то и проблема: UUIDs хороши для компьютеров, поскольку дают простую и надежную идентификацию, но недружественны к людям. Компромиссом будет использование ярлыков файловой системы. Если ваш раздел root находится на **/dev/sda1**, можно запустить

```
sudo e2label /dev/sda1 root
```

и изменить спецификацию устройства в **/etc/fstab** с **"UUID=youllneverreadorrememberthis"** на **"LABEL=root"**. В отличие от многих операций в файловой системе root, эту совершенно безопасно выполнять на работающей системе. Она вступит в силу после перезагрузки. Желая перемещать диски с одного компьютера на другой, выберите не такой общепринятый ярлык, как "root".

> «Свободная память ничего не делает, а могла бы повысить производительность.»

Копирование жесткого диска

В **Как мне клонировать весь жесткий диск целиком? Я использую Ubuntu, но на диске есть еще и Windows.**

О Клонировать диск, на котором работает система — задача рискованная. ОС может изменить файлы на диске, когда вы будете пытаться ее клонировать, и, вероятно, так и сделает. Самый простой и самый безопасный способ сделать клон — с помощью Clonezilla, специального дистрибутива live CD. Запишите ISO на CD и загрузитесь с него. Затем используйте меню Clonezilla, чтобы сделать клон вашего жесткого диска — или на другом диске, или в файле образа.



Для копирования всего жесткого диска обычно самым первым выбором будет Clonezilla.



Программные глюки

Программы куролесят? Или их нет? Мы поможем.

Установка программ

В **Я — новичок в Linux, и не могу понять, как установить программы. В Windows я захожу на сайт программы, скачиваю файл .exe и запускаю его. А здесь мне что делать?**

О Установка программ в Linux делается иначе. Обычно программы предоставляются в виде исходного кода, и отдельные дистрибутивы создают из него программные пакеты. Это гарантирует, что программа совместима с остальными программами в вашем дистрибутиве, а также установку из самого дистрибутива. Найдите программу под названием *Software Manager* или наподобие: она позволит вам найти и установить все требуемые программы. Кроме того, она позаботится о зависимостях (о других программах, нужных той, которую вы устанавливаете) и уведомит вас о наличии обновлений. Ни вам, ни программам не понадобится проверять web-страницы, чтобы «отзвониться».

Возврат к источникам

В **Я хочу установить программу, но для моего дистрибутива нет пакета, а сайт программы предлагает только исходный код. Как мне ее установить?**

О Конкретный метод установки зависит от программы. Первым шагом всегда будет распаковать tar-архив или файл ZIP и найти файл под названием **Readme**, **Install** или нечто в этом роде. В нем будет объяснение, как установить программы. Для 90% проектов используются стандартная система *autotools*, которая требует для компиляции и установки программ трех команд, запускаемых из директории, созданной при распаковке tar-архива:

```
./configure
make
sudo make install
```

Первая команда проверяет, есть ли в вашей системе все необходимые зависимости, и настраивает среду компиляции в соответствии с вашей системой. Вторая компилирует исходный код, чтобы программа работала на вашем компьютере, а третья команда устанавливает эти программы в нужное место и занимается окончательной настройкой.

Чтобы это сделать, у вас должна быть установлена стандартная среда компиляции. В большинстве дистрибутивов есть пакеты под названием *build-essentials*, или нечто подобное; они установят все, что нужно для компиляции и установки программ из исходника.

После установки исходную директорию не удаляйте. Если впоследствии вы решите удалить программу, вернитесь туда и запустите

```
sudo make uninstall
```

› **Не выискивайте в Сети программы из сомнительных источников. Большая часть того, что вам надо, есть в менеджере ПО вашего дистрибутива.**



Нет DVD-привода

В **У меня есть ISO-образ, который я могу записать на CD/DVD, но я хочу использовать его с компактным ноутбуком без DVD-привода. Можно ли мне использовать вместо него USB-брелок?**

О Да, можно. Точный метод зависит от типа ISO-образа. Некоторые дистрибутивы сегодня выходят с так называемыми гибридными ISO-образами, которые работают на любом носителе. Вы записываете их на DVD с помощью любой программы прожига DVD, или скопируйте файл на USB-брелок с помощью **dd**:

```
sudo dd if=distro.iso of=/dev/sdb bs=4k
```

где **if** — копируемый файл, а **of** — пункт назначения (**bs=4k** солидно ускоряет процесс, но не является обязательным). Предупреждение: **dd** делает именно то, что вы ей скажете, без всяких любезностей/досадностей типа «Вы уверены, что...», поэтому загодя проверьте правильность указанного диска назначения.

Как узнать, что у вас гибридный ISO? Об этом вам может сообщить сайт дистрибутива, или проверьте это с помощью **fdisk**:

```
sudo fdisk -l distro.iso
```

Если строка под строкой Device включает имя ISO-образа, значит, у вас гибрид. Если нет, конвертируйте его через *isohybrid*. Возможно, сначала придется установить пакет *syslinux*, тогда запустите

```
isohybrid distro.iso
```

Запустите из оболочки

Если программа отказывается работать из программы запуска приложений рабочего стола, или работает, но потом выдает отказ или иные проблемы, попробуйте запустить ее из терминала. Это не заставит ее волшебным образом работать правильно, но большинство программ, если что-то идет не так, отправляют сообщения в терминал, так что у вас может появиться подсказка, или, по крайней мере, информация, которую можно ввести в поисковую машину. Здесь предполагается, что вы знаете имя команды, чтобы запустить ее из терминала, а это имя часто являет собой название программы, написанное маленькими буквами; то есть, чтобы запустить *Firefox*, вам надо ввести **firef**[TAB], чтобы найти правильное имя. Если вы не можете угадать имя таким образом, выведите список программ, которые пакет установил, в свой путь команд. Например, для производных Debian командой будет:

```
dpkg -L firefox | grep bin/
```

Это заглавная буква L — маленькая буква используется для поиска файлов пакетов вместо установленных пакетов. Эквивалентом для RPM является

```
rpm -ql | grep bin/
```

Что касается **grep**, тут вместо полного списка пакетов показываются только файлы, установленные в одну из директорий ***bin**.

Теперь fdisk должен показать его как гибрид, и вы можете сделать **dd** на USB-брелок.

Тормоз

В Такое ощущение, что при работе моего компьютера одно из ядер CPU стреножено. Что происходит?

О Самая вероятная причина в том, что некий процесс потребляет слишком много ресурсов, а узнать это поможет программа под названием *top*. Запустите ее в терминале, и она покажет вам сводку полного использования ресурсов системой и список программ в порядке использования ими CPU. Третья строка покажет вам общее использование CPU, и это будет выглядеть примерно так (на слабо загруженной системе):

```
%Cpu(s): 0.7 us, 0.2 sy, 0.0 ni, 99.1 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
```

Первые три цифры показывают, какая часть мощности CPU используется для пользователя, системы и неважных задач. Неважная задача — это задача, запущенная под низким приоритетом; ей выдается CPU, только если никакой другой задаче он не нужен. Четвертая цифра — «простой», то есть неиспользуемое время CPU. Если она близка к нулю, то вы теперь знаете, почему ваш компьютер работает медленно — посмотрите на список задач, чтобы выяснить, которая потребляет больше всего времени CPU. Если у вас многоядерный CPU, то нажатие на 1 позволит вам переключиться с отображения общего использования на показатели для каждого ядра. Цифра у **wa** (waiting — ожидание) тоже важна: она показывает, сколько времени CPU ожидает ввода/вывода, и может сильно сказываться на операциях, интенсивно использующих диск. Если она обычно достигает высоких показателей, у вас может иметь место проблема с диском — или ошибки при чтении, или почти заполненная файловая система, которая становится сильно фрагментированной.

Посмотрите на строку памяти. Если память переполнена и используется swap, то вы теряете производительность. Список задач определяется использованием CPU, но вы можете нажать F, чтобы вывести список областей. Перемещайте курсор по области, которую вы хотите рассортировать, и нажмите S, а затем Q. И хотя это звучит довольно запутанно, но окно, которое открывает F, позволяет делать намного больше, чем просто изменять порядок сортировки. Если памяти не хватает, оно поможет вам выявить прожорливые программы.

Потерянные пароли

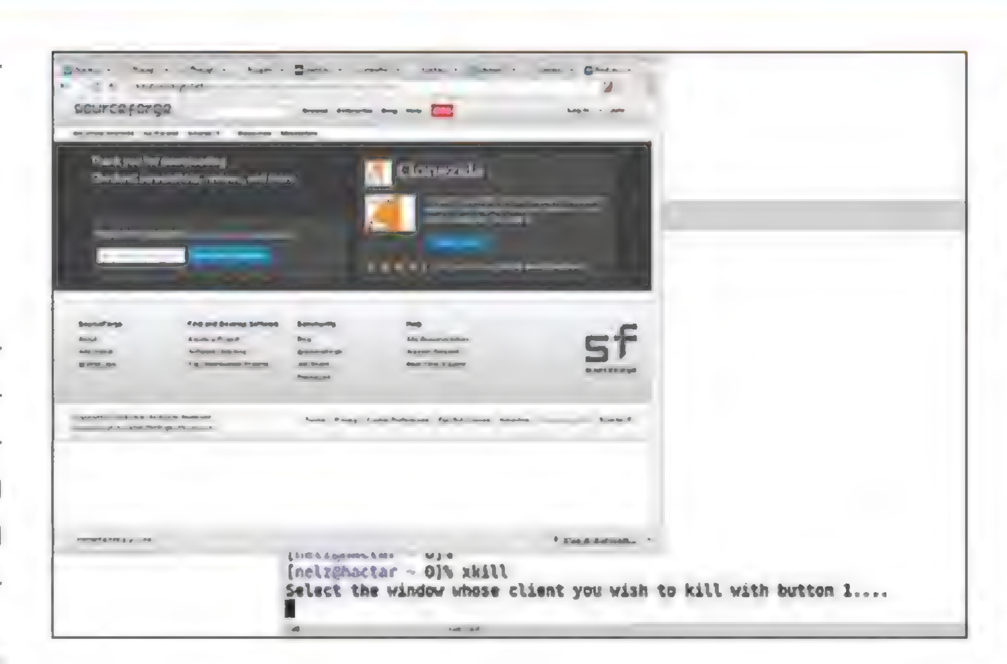
В Я недавно переустановил Linux, и думал, что у меня остался тот же пароль, но я не могу войти в систему с его помощью. Есть ли возможность найти или переустановить пароль?

О Новый пароль можно задать, загрузившись с live CD/DVD — или со специального диска аварийного восстановления системы, например, System Rescue CD или собственного диска вашего

```
top - 12:32:02 up 1 day,  3:55, 10 users,  load average: 0.88, 0.81, 0.64
Tasks: 329 total,   4 running, 325 sleeping,   0 stopped,   0 zombie
%Cpu(s): 11.0 us, 13.6 sy,  0.0 ni, 74.5 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem:  16426484 total, 16298072 used,   128412 free,     1064 buffers
KiB Swap: 50331636 total,       64 used, 50331572 free.  9982416 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 1501 nelz      20   0  527856  61644  42968 R  91.0  0.4   4:37.09 konsole
 9612 nelz      20   0    6008    640    560 R  52.5  0.0   0:09.47 yes
 4692 root      20   0       0      0      0 S  16.9  0.0   0:01.85 kworker/0:1
 9618 root      20   0       0      0      0 S  12.0  0.0   0:00.60 kworker/1:0
 1876 root      20   0       0      0      0 S   5.6  0.0   0:01.19 kworker/3:25
  900 root      20   0  328960 139584  88972 S   4.3  0.9  22:40.02 X
 8958 root      20   0       0      0      0 S   3.0  0.0   0:00.48 kworker/5:9
 9480 nelz      20   0 2581628 1.356g  17316 S   2.3  8.7   1:09.18 qemu-system-x86
 8567 root      20   0       0      0      0 S   2.0  0.0   0:00.16 kworker/7:10
 9616 nelz      20   0  264480  37280  31612 S   2.0  0.2   0:00.15 ksnapshot
  769 root      20   0  451260  39244  11724 S   1.7  0.2  45:53.79 motion
 1087 nelz      20   0 6528944 269324  99928 R   1.7  1.0  25:54.12 plasma-desktop
 4676 root      20   0       0      0      0 S   1.7  0.0   0:01.31 kworker/2:2
 6851 root      20   0       0      0      0 S   1.7  0.0   0:00.14 kworker/3:1
 6909 root      20   0       0      0      0 S   1.3  0.0   0:00.07 kworker/2:11
 8027 root      20   0       0      0      0 S   1.3  0.0   0:00.17 kworker/1:5
 9447 root      20   0       0      0      0 S   1.3  0.0   0:00.19 kworker/4:0
 1726 root      20   0       0      0      0 S   1.0  0.0   0:05.68 kworker/0:7
16622 nelz      20   0  917700 205456  53264 S   1.0  1.3  14:20.68 chrome
16633 nelz      20   0  934616 212564  53792 S   0.7  1.3  15:04.58 chrome
16685 nelz      20   0  882012  30004  21852 S   0.7  0.2   5:15.12 vlc
 1494 nelz      20   0  218124  29236  24160 S   0.3  0.2   1:41.05 kfilebox
```

› ***Top*** **дает сводку по ресурсам, используемым вашей системой, а также список программ, потребляющих больше всего ресурсов.**



› ***Xkill*** **закроет первое окно, по которому вы щелкнете, вместе с его процессом, так что будьте внимательны.**

дистрибутива. Первый шаг — определить раздел, где установлен ваш дистрибутив (это файловая система root для дистрибутива; ваша директория **home** тут ни при чем). Если вы загрузили графический рабочий стол, можете просмотреть имеющиеся диски в поисках диска подходящего размера, пока не найдется тот, что содержит директории типа **bin**, **etc** и **lib**. Когда у вас будет смонтирован правильный раздел, скажем, в **/mnt/myroot**, вы сможете использовать **chroot**, чтобы его исправить:

```
sudo chroot /mnt/myroot /bin/bash
```

Это направит директорию root по указанному вами пути и запустит следующую за тем команду, переключив вас в root вашей ОС, и оболочка *Bash* запустится от имени root. Теперь вы находитесь внутри своей установки и можете изменить свой пароль:

```
passwd yourusername
```

> «В дистрибутивах обычно предусмотрена установка программ и компиляция.»

У вас дважды спросят новый пароль, и затем он запишется на диск. Нажмите Ctrl+D для выхода из **chroot**, перезагрузитесь с жесткого диска, и вы теперь должны получить возможность заходить в систему.

Окноубийство

В Я работаю с самыми новыми программами, и иногда они дают сбои, но окна при этом остаются открытыми. Как закрыть окно и такую программу, чтобы я смог запустить их снова?

О Для графических программ простейшим ответом будет утилита *xkill*, обычно установленная по умолчанию. Когда вы запустите *xkill*, курсор мыши превратится в череп с перекрещенными костями; следующее окно, по которому вы щелкнете, будет принудительно закрыто вместе с его программой, так что внимательно смотрите, где щелкать.

Есть и альтернативы командной строки. **kill** берет ID процесса (PID) как аргумент и убивает процесс. **Killall** делает то же, но работает с именами программ; будьте внимательны, если у вас может работать более одного экземпляра программы. Обе команды по умолчанию отправляют сигнал TERM, а тот недвусмысленно предлагает программам закрыться. Если это не сработало, добавьте к аргументам **-KILL** — отправится более сильный сигнал KILL:

```
killall -KILL имяпрограммы
```

Если требуется **kill**, есть разные способы найти PID. Если процесс использует много CPU или памяти, он будет показан вверху, и можно закрыть его там, нажав K. Иначе, можно применить **pgrep**:

```
pgrep -fl имяпрограммы
```

Опции **-fl** означают, что вы видите командную строку полностью и убиваете именно тот процесс, который надо.

Соединение

А теперь займемся проблемами связи и Интернета.

Потеря скорости

В Мое интернет-соединение кажется очень медленным – скачивание файлов тянется бесконечно. Следует ли мне решить эту проблему в Linux или изменить настройки моего роутера?

О Настройки роутера лучше не трогать, пока не попробуете все остальное — вы можете испортить совершенно рабочую настройку. Первый шаг — проверка, действительно ли теряется скорость, через такой сервис, как http://speedtest.net. Далее, выясните, не используется ли ваш канал чем-нибудь в фоне. Хорошей программой для этого будет *nethogs*, которую вам, возможно, придется установить из вашего менеджера пакетов:

```
sudo nethogs eth0
```

Вы получите отображение в стиле *top* того, что занимает вашу полосу пропускания. Если у вас беспроводное соединение с роутером, попробуйте проводное соединение, чтобы отсечь проблемы с Wi-Fi. Увидев разницу, попробуйте протестировать в разных местоположениях.

Существуют также шансы, что канал перегружен, поскольку используется также другими беспроводными роутерами, расположенными по соседству, и эту проблему можно решить, переключившись на другой беспроводной канал. Вы сможет увидеть, какие беспроводные каналы используются, с помощью команды

```
sudo iwlist wlan0 scan I grep -e Address: -e Channel: -e ESSID:
```

Iwlist выдает кучу информации по каждой обнаруженной ею точке доступа, отсюда необходимость фильтровать ее вывод с помощью **grep**. Вы можете также использовать что-нибудь вроде *Wi-Fi Analyser* на Android, который графически отображает канал и мощность всех имеющихся точек доступа, позволяя определить качество сигнала и занятость канала.

Есть также высоконаучный тест с помощью телефона: если линия шумит, значит, у вас будет меньшая скорость, и вам нужно сообщить о проблеме своему провайдеру. Если же все вышеперечисленные тесты ничего не показывают, пообщайтесь со своим интернет-провайдером. И уж только потом беритесь за настройки роутера.



› ***Wi-Fi Analyzer*** **на Android показывает точки доступа в зоне досягаемости и используемые ими каналы, и вы сможете настроить свою так, чтобы избежать конкуренции.**

Потоковое видео

В Когда я пытаюсь посмотреть видео на Amazon Prime или Netflix (у меня есть учетные записи на обоих сервисах), я получаю сообщение, что мой компьютер не поддерживается. Могу ли я заставить их верить, что я использую Windows? Смогу ли я тогда смотреть то, за что платил?

О Эти сервисы используют *Silverlight*, проприетарную технологию от Microsoft, для рендеринга видео и использования DRM. *Silverlight* для Linux нет, но есть решение под названием *Pipelight* — это плагин браузера, который, в свою очередь, использует *Wine* для запуска *Silverlight*. Это требует подлатанной *Wine*, но все нужные вам пакеты в большинстве дистрибутивов имеются. Пользователи дистрибутивов на основе Ubuntu могут добавить PPA и все установить командами

```
sudo add-apt-repository ppa:pipelight/stable
sudo apt-get update
sudo apt-get install pipelight-multi
```

Это добавит PPA Pipelight и затем установит программы. Инструкции для других дистрибутивов находятся на http://bit.ly/PipelightInstall. Настройте их с помощью

```
sudo pipelight-plugin --create-mozilla-plugins
pipelight-plugin --enable silverlight
```

Скачаются нужные программы *Silverlight* и настроятся плагины для вашего браузера. Установки *Pipelight* недостаточно, чтобы смотреть видео; вам по-прежнему будет отказано в доступе, потому что ваш браузер сообщит сайту, что вы работаете в Linux. Установите одно из расширений управления User-Agent для вашего браузера: например, user-agent-overrider или uacontrol в *Firefox* или useragent-switcher-for-chrome, если вы используете *Chrome*. Затем

Предпоследнее средство

Когда кажется, что ваш компьютер полностью заблокирован, и даже яростные удары по клавиатуре не дают никакого результата, то самым крайним средством будет отключить питание, со всем вытекающим риском потери и повреждения данных. Однако есть еще одно средство, которое вы можете попробовать: клавиши Magic SysReq.

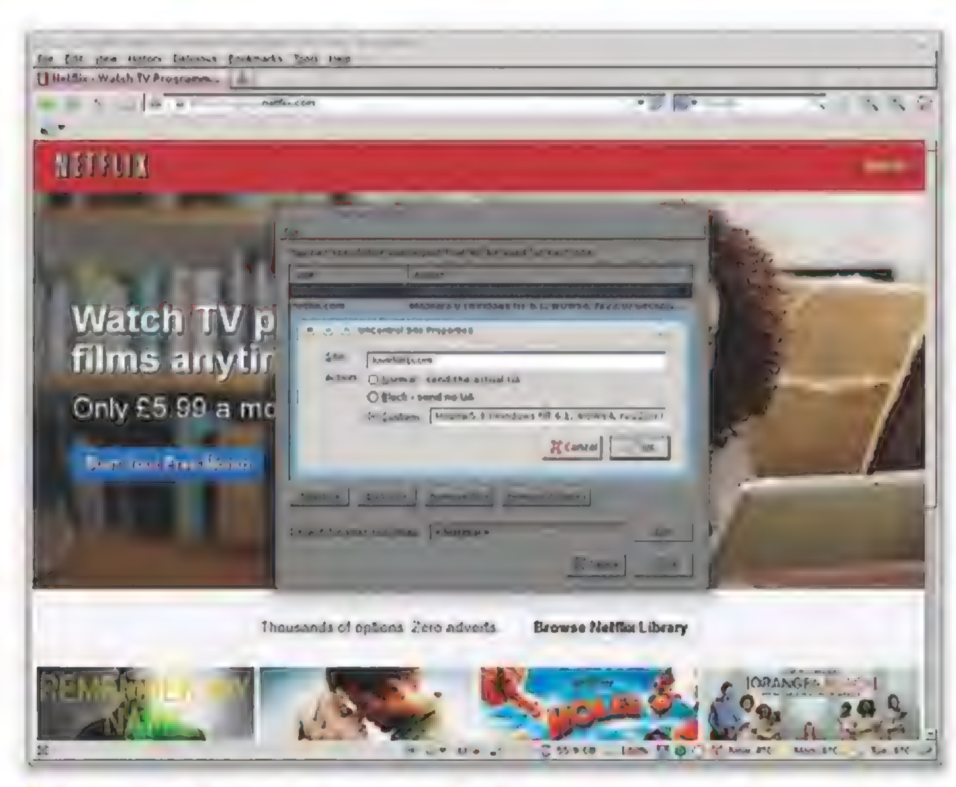
Клавиши Magic SysReq — это специальные комбинации клавиш, которые довольно мягко выключают зависшую систему. Их название объясняется тем, что они используют клавишу SysReq, которая иногда помечена, как PrintScr; и они воистину волшебные.

Даже если ваша клавиатура вообще не работает, они сработают, потому что ядро реагирует на нажатие этих клавиш непосредственно, без всякого вмешательства программ из области пользователя.

Удерживая Alt и SysReq, нажмите по очереди на каждую из этих клавиш:

- **R** — перезапускает клавиатуру.
- **E** — выходит из всех оставшихся работающих программ по сигналу KILL.
- **I** — прерывает все работающие программы сигналом TERM.
- **S** — синхронизирует все файловые системы.
- **U** — размонтирует все файловые системы и заново монтирует их только на чтение.
- **B** — перезагружает.

Есть разные вариации для запоминания этих клавиш — мнемонические сочетания — и некоторые из них труднее запомнить, чем сам порядок нажатия этих клавиш; в одном упоминаются слоны. Простейшим для запоминания мне показалось прочесть их в обратном порядке (получится слово BUSIER — *англ.* более занятой), но больше всего мне нравится анаграмма "Reboot Even If System Utterly Broken [Перезагрузит, даже если система капитально поломалась]".



› Запустите Netflix, установив *Pipelight* и дав своему браузеру команду притвориться, что он работает в Windows.

используйте его, чтобы настроить ваш User Agent на “Firefox для Windows” и наслаждайтесь своим потоковым видео.

Монтируем Android

В Я купил планшет Nexus 7 Android и очень им доволен. Но — всегда найдется какое-нибудь «но», правда? — я не могу монтировать его как устройство USB. Мой телефон на Android при подключении его к USB-порту на моем компьютере выдает сообщение и позволяет монтировать себя как устройство хранения, а вот планшет так работать не хочет. Почему они убрали такую разумную систему и как мне теперь перекинуть на него файлы?

О Проблема с USB-хранилищами в том, что файловую систему можно монтировать только на одной ОС за раз. Это означает, что прежде чем вы сможете монтировать такое на своем компьютере как устройство массового хранения, ее надо размонтировать на телефоне/планшете. Поэтому у вас на телефоне будут появляться предупреждения о том, что нужно остановить приложения, использующие SD-карту. Ситуация еще хуже на таких устройствах, как Nexus 7, где нет съемного устройства хранения — вы не можете размонтировать файловую систему, которую в данный момент использует ОС. Более новые устройства Android избегают этого, используя MTP (Media Transfer Protocol). Есть несколько опций монтирования устройств MTP в Linux, все они реализованы в качестве файловых систем FUSE. Я предпочитаю jmtpfs. Чтобы смонтировать первое обнаруженное устройство — обычно у вас подсоединено только одно за раз — передайте jmtpfs имя точки монтирования. Директория должна существовать и быть записываемой пользователем при запуске команды:

```
jmtpfs ~/nexus
```

Затем вы должны получить доступ к содержимому устройства, продолжая им пользоваться. Закончив, размонтируйте устройство командой

```
fusermount -u ~/nexus
```

Удаленные рабочие столы

В Я хочу, чтобы у меня был доступ к рабочему столу одного компьютера с другого, как в функции удаленного рабочего стола в Windows.

О Здесь у вас две опции. Для полного рабочего стола вы можете использовать *VNC*. Он часто установлен по умолчанию; если нет, установите пакет *TightVNC* или *VNC* (первый должен увеличить скорость в медленных сетях). Вы запускаете сервер на компьютере, который хотите просмотреть; в первый раз вам предложат задать пароль. Затем запустите клиент на другом компьютере, дав ему адрес первого компьютера.

Но если вы хотите только запустить одну программу с удаленного компьютера на локальном рабочем столе, *VNC* будет многовато, а X-дисплей в Linux уже позволяет передавать данные по сети. Вам нужно настроить SSH на удаленном компьютере — просто установите обычным способом пакет *ssh-server*. Затем проверьте соединение с локального компьютера с помощью

```
ssh user@hostname
```

У вас спросят пароль, а затем откроется сессия терминала на удаленном компьютере — вы можете убедиться, что находитесь на удаленном компьютере, запустив **uname -a**. Нажмите Ctrl+D, чтобы выйти, затем попробуйте запустить графическую программу, вот так:

```
ssh -Y user@hostname someprogram
```

Перед вами должно открыться ее окно. В зависимости от того, как ваш дистрибутив настроил свой сервер SSH, вам может показаться, что **-X** лучше, чем **-Y**. Если не работает ни то, ни другое, проверьте **/etc/ssh/sshd_config** на компьютере, где работает сервер, и убедитесь, что он содержит

```
X11Forwarding yes
```

Между прочим, если вы используете одно и то же имя пользователя на обоих компьютерах, вы можете пропустить его, и просто выполнить

```
ssh -Y hostname progname
```

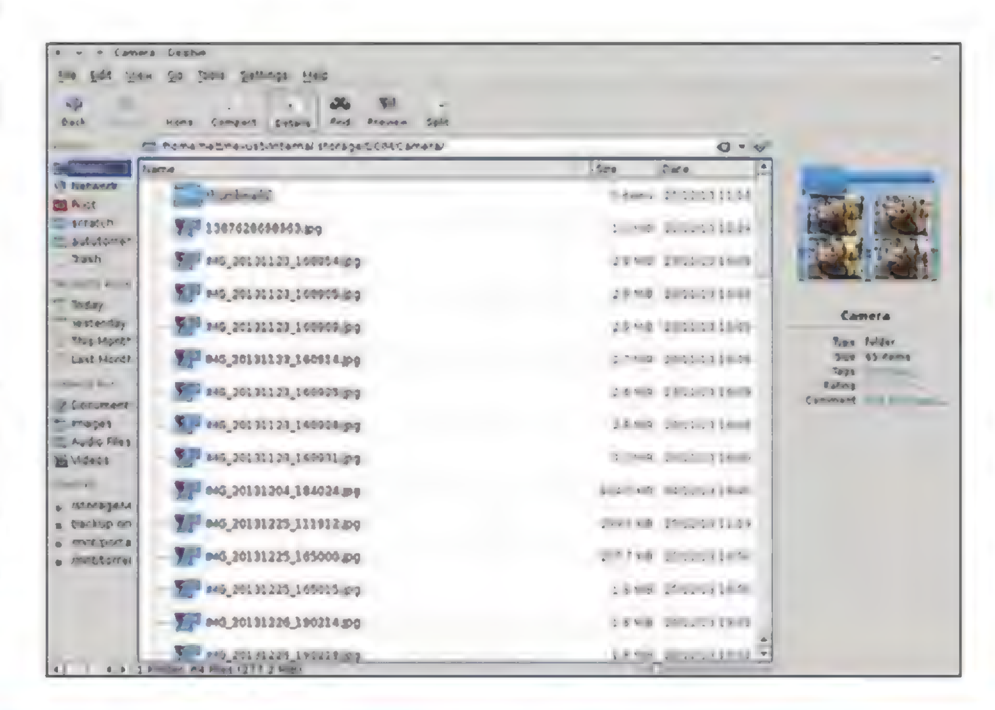
Какой у меня адрес?

В У меня Raspberry Pi, который я использую без клавиатуры или экрана. Я подключил его к своей сети, но не знаю его IP-адреса, поэтому не могу сделать SSH.

О Ваше устройство получает свой IP-адрес с сервера DHCP; часто вы можете настроить сервер так, чтобы он давал особый адрес имени хоста. Подключите SD-карту вашего Pi к своему компьютеру и отредактируйте на ней **/etc/hostname**, чтобы настроить свое имя хоста, затем настройте свой сервер DHCP; обычно это делается в роутере, чтобы дать этому имени хоста установленный адрес. В качестве альтернативы установите *netdiscover* (http://bit.ly/Netdiscover) и запустите его так:

```
sudo netdiscover -i eth0 -r 192.168.1.0/24
```

где **-i** — ваш сетевой интерфейс, а **-r** — диапазон IP, используемый вашей сетью. Он перечислит все хосты вашей сети. К сожалению, Pi отображается только как “Unknown vendor [Изготовитель неизвестен]”, так что запустите его с и без Pi, чтобы увидеть появляющийся адрес. Если у вас устройство Android, установите *Fing*, который сделает все то же самое в графическом виде и распознает Pi. Получив адрес, вы сможете использовать на нем SSH и запустить **ipconfig** для получения MAC-адреса, который вы сможете использовать, чтобы сообщить своему серверу DHCP, какой адрес ему присвоить. Raspbian при загрузке выдает IP-адрес из **/etc/rc.local**. Вы можете поменять это на отправку его вам по электронной почте.



› MTP — новый способ соединения с устройствами Android; благодаря ему вы можете продолжать использовать устройство, пока оно монтируется на вашем настольном ПК.

Проблемы рабочего стола

Убедимся, что среда, которая маячит у вас перед глазами, работает.

Нет рабочего стола

В Я применил обновления системы по уведомлению своего рабочего стола Ubuntu. После следующей перезагрузки передо мной был черный экран, строка приглашения, мерцающий курсор — и больше ничего. Я смог войти под своим именем пользователя, но рабочего стола по-прежнему не было.

О Это произошло потому, что обновления системы установили новое ядро, которое стало настройкой по умолчанию после перезагрузки. Если вы используете видеокарту со сторонними драйверами, например, Nvidia, нужно переустановить драйверы, чтобы вы смогли вернуть свой рабочий стол. Если вы устанавливали драйверы через *Software Center*, он должен позаботиться об их переустановке; а если вручную, то это будет на вашей совести, а драйверам видеокарты для работы нужен модуль ядра. Если вы используете драйвер с открытым кодом, то модуль включен в ядро, поэтому процесс переключения с одного ядра на другое прозрачен. С двоичными драйверами модуль компилируется и устанавливается программой установки драйверов, и для нового ядра вам нужно сделать это повторно. Хорошая новость в том, что вы можете перезагрузиться, выбрать предыдущее ядро в меню згрузки, и ваш рабочий стол вернется. Затем вы можете снова установить драйверы точно так же, как вы делали в прошлый раз.

Пользователь неправ, замените его

В Программа *XYZ* начала выдавать сбои, она не работает, как раньше. Я попытался ее переустановить, но это не помогло.

О Не обновляли ли вы недавно эту программу или одну из библиотек, которыми она пользуется? Загляните в лог-файлы своего менеджера пакетов, чтобы выяснить, что недавно было установлено (пользователи дистрибутивов на базе Ubuntu или Debian могут заглянуть в **/var/log/apt/history.log**). Если ничего не менялось, то это, скорее всего, проблема настройки. Переустановка пакета не повлияет на настройки пользователя, которые хранятся где-то в вашей домашней директории. Простейший способ проверить это — создать нового пользователя, войти в систему под его именем и попробовать ту же программу, что отправит вас к настройке по умолчанию. Если проблема исчезла, значит, это проблема с вашей настройкой.

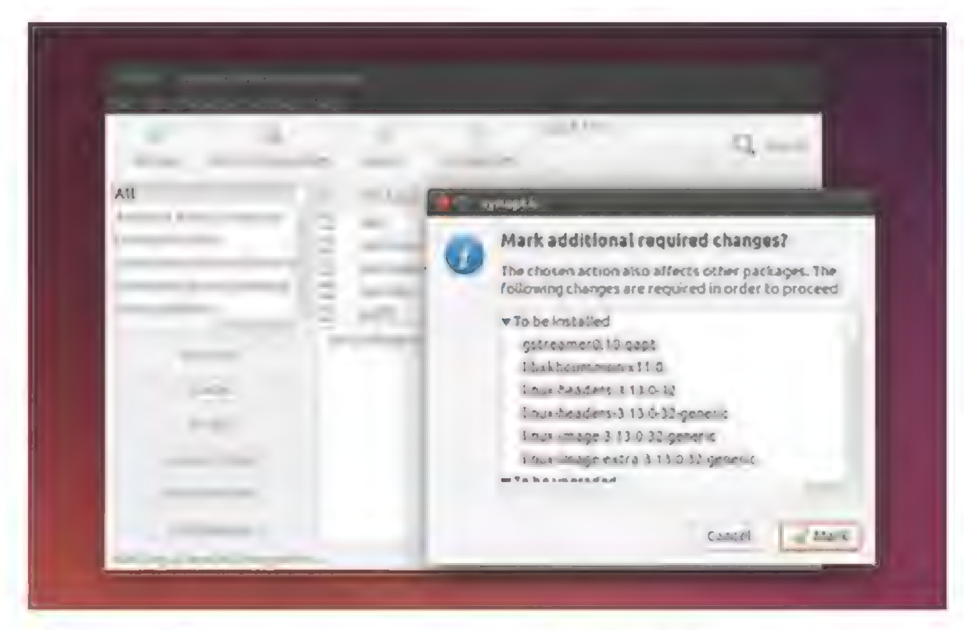
Следующий шаг — выяснить, где программа хранит свою настройку. Программы KDE хранят свои настройки в **~/.kde4/share/config**, но большинство других программ используют **.config**, **.local** или пишут директорию своей настройки файлов в вашей домашней директории. Простой способ найти файлы, измененные программой — использовать команду **find**. Сначала создайте файл с отметкой текущего времени

```
touch ~/now
```

Затем запустите свою программу, измените настройку и выйдите из нее. Теперь вы найдете в своей домашней директории все файлы, которые были изменены, с помощью

```
find ~ -newer ~/now
```

Обнаружив файл или директорию настройки программы со сбоями, вам надо будет ее переименовать и запустить программу снова, чтобы вернуться к стандартной настройке. И не забудьте о создании нового пользователя без всяких настроек — в будущем это может сэкономить вам уйму времени.



Разумно будет проверить список обновляемых пакетов: если среди них есть новое ядро, после перезагрузки оборудование может перестать работать, пока вы не переустановите драйверы.

Не принимается логин

В У мня постоянно возникают проблемы, когда я пытаюсь заставить Ubuntu принять мой пароль, который правилен, я точно это знаю. Изначально я настроил автоматическую установку; тем не менее, передо мной появился экран приглашения, но мой пароль не был принят. А если я загружаюсь в режиме восстановления, то с моим паролем, тем же самым, все в порядке!

О Проблема не в пароле, иначе вы бы не смогли войти в консоль восстановления. Что-то другое не дает вам войти в среду рабочего стола, именно поэтому вы и видите экран приглашения, хотя и включили автоматический вход в систему. Он пытается автоматически войти в систему, у него не получается, и он возвращается к экрану приглашения. Обычно такое бывает из-за

Где еще найти помощь

Итак, вы попытались решить проблему, но она не исчезла. Куда идти дальше? Первый шаг — вставить любое полученное сообщение об ошибке в поисковик. Есть целый ряд мест, куда можно обратиться за помощью. Если ваша проблема связана с дистрибутивом, попробуйте спросить на web-форумах дистрибутива и через списки рассылки. Если она более общая, обратитесь на http://linuxquestions.org или в *Linux Format* (www.linuxformat.com/forums).

На IRC болтаются многие разработчики, и там можно быстро получить ответ. Что бы вы ни спросили, относитесь к любым советам с осторожностью. Конечно, бывают люди, получающие извращенное удовольствие от предоставления вредных советов, но гораздо опаснее люди благонамеренные, которые искренне пытаются помочь, но не поняли вашу проблему. По возможности, подождите консенсуса с советами, или разузнайте об опыте советчика, прежде чем запускать любые команды, чреватые повреждением системы.

Действительно ценный источник советов — Группы пользователей Linux (LUG). У большинства из них есть списки рассылки, а также бывают встречи. Если расспросы в списке рассылки не решили вашу проблему, а проблема на портативном устройстве, вы можете взять его с собой на встречу для получения практической помощи.

неправильного права собственности на файлы, в которых рабочий стол должен делать записи. Вы могли неумышленно передать права собственности на критически важный файл root — обычно это происходит в результате попытки решить какую-то проблему.

Снова загрузитесь в режим восстановления и войдите под именем пользователя, у которого есть проблемы, и затем запустите эту команду, чтобы перезапустить права собственности на все файлы в вашей домашней директории и вернуть их вам:

```
sudo chown -R ${USER}: ~
```

Команда **chown** используется для изменения прав собственности [CHange OWNership] на файлы. Опция **-R** велит рекурсивно изменить все файлы в выбранном пути. Переменная среды ${USER} расширяется до вашего имени пользователя (именно поэтому вам надо войти от имени пользователя, у которого возникли проблемы, чтобы все заработало). Двоеточие (:) после имени пользователя велит **chown** изменить также права собственности на группу по умолчанию для этого пользователя. Наконец, оболочка расширяет **~** до пути к вашей домашней директории. Вам надо запустить **chown** с помощью **sudo**, поскольку для изменения прав собственности на все, что вам не принадлежит, нужны разрешения root, а к этой-то категории и относятся интересующие нас файлы. Если по-прежнему ничего не получилось, загляните в **.xsession-errors** в вашей домашней директории.

Заблокированы устройства

В У меня не всегда получается извлечь DVD или USB-брелок, я получаю сообщение: «Не удалось выполнит действие: устройство используется». Как мне обойти эту проблему без перезагрузки, которая помогает?

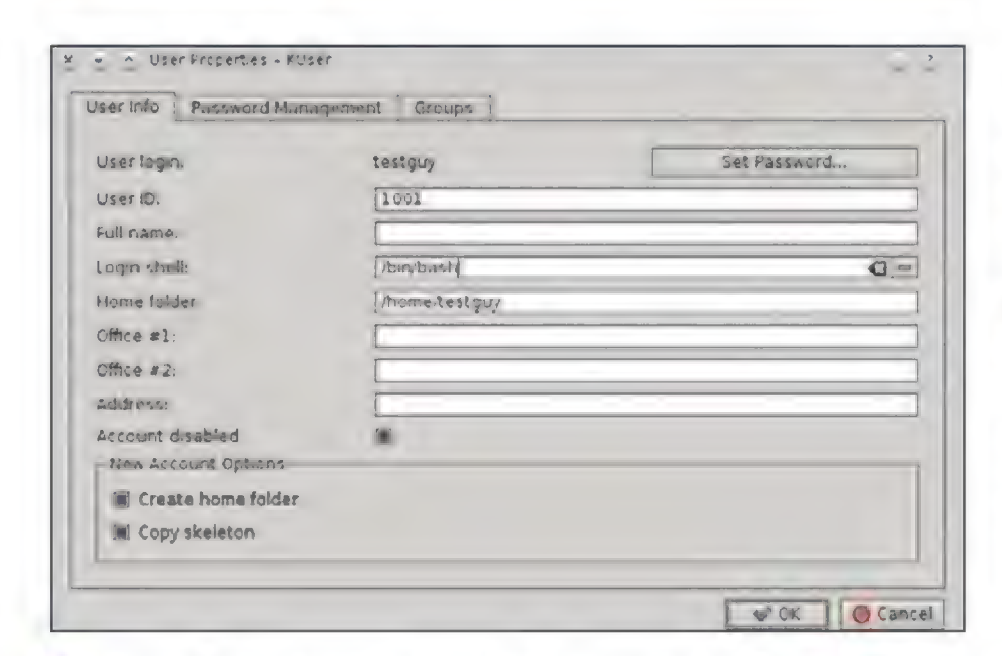
О Какая-то программа где-то заблокировала файл или директорию в файловой системе, которую вы хотите размонтировать, поэтому операционная система отказывается делать то, что вы хотите. Команда **lsof** перечисляет все открытые файлы, включая директории и узлы устройств. Пропустив этот список через **grep**, вы сможете найти виновный процесс, например:

```
# lsof | grep /media/cdrom
bash 15243 nelz cwd DIR 11,0 6144 1856 /media/cdrom
```

Здесь показано, что *Bash*, работающий под моим пользователем, заблокировал **cwd** в этой директории. **cwd** — Change Working Directory — иными словами, у меня открыта оболочка, которая отображает директорию CD. Если просто задать команду **cd** в этой оболочке, чтобы вернуться в мою домашнюю директорию, она снимет блок и позволит размонтировать CD. Выяснив, какая программа блокирует устройство, вы сможете с ней разобраться.

Альтернативной командой будет **fuser**, она просто возвращает ID процесса программы, блокирующей файл:

```
# fuser /media/cdrom
```



› Создание нового пользователя с настройками по умолчанию — простой способ выяснить, является причиной проблемы сбой программы или нет.



› Если вы опять оказались перед экраном приглашения, причина может быть в том, что вы напутали с разрешениями файлов или правами собственности.

```
/media/cdrom: 15243c
```

Вы увидите, что это тот же PID, который выдал **lsof**. Это может выглядеть менее удобным, и в данном случае так и есть, потому что мне нужно только найти оболочку и изменить директории, но у **fuser** есть в рукаве козырь. Если использовать

```
fuser -k /media/cdrom
```

она найдет PID и отправит сигнал KILL, немедленно останавливая программу и снимая блокировку, что очень удобно, если программа, выдавшая отказ, оставила блокировку на устройстве. Это безопаснее, чем удалить устройство, не размонтируя его, что может привести к потере данных.

Где же этот файл?

В Не знаю, в чем дело, в моем возрасте или в чем-то еще, но у меня есть привычка сохранять файлы и потом забывать, где я их сохранил, и в итоге передо мной открывается перспектива просматривать многие гигабайты файлов в своей домашней директории. Можете помочь человеку с хронической забывчивостью?

О Помимо таких вещей, как семантический рабочий стол KDE, который индексирует все файлы, которые вы создаете, есть и несколько общих опций. Первая — это **find**, которая (о чудо!) находит файлы на основе разных критериев. Например, если вы знаете приблизительно, где вы сохранили файл, вы можете использовать **-mtime** для поиска последней измененной временной метки файла:

```
find ~ -type f -mtime 0
find ~/documents -type f -mtime +7 -mtime -14
```

Первая находит все файлы в вашей домашней директории, измененные за последние 24 часа; вторая ищет в директории документов файлы, которым больше недели, но меньше двух недель. Нам нужна **-type f**, чтобы не искать совпадений по директориям. Можно также искать совпадения по именам файлов, чтобы найти электронные таблицы, измененные за последнюю неделю:

```
find ~ -mtime -7 -name '*.ods'
```

Если это текстовый файл и вы знаете часть содержимого файла, можете использовать **grep**:

```
grep -rl "some text" ~
```

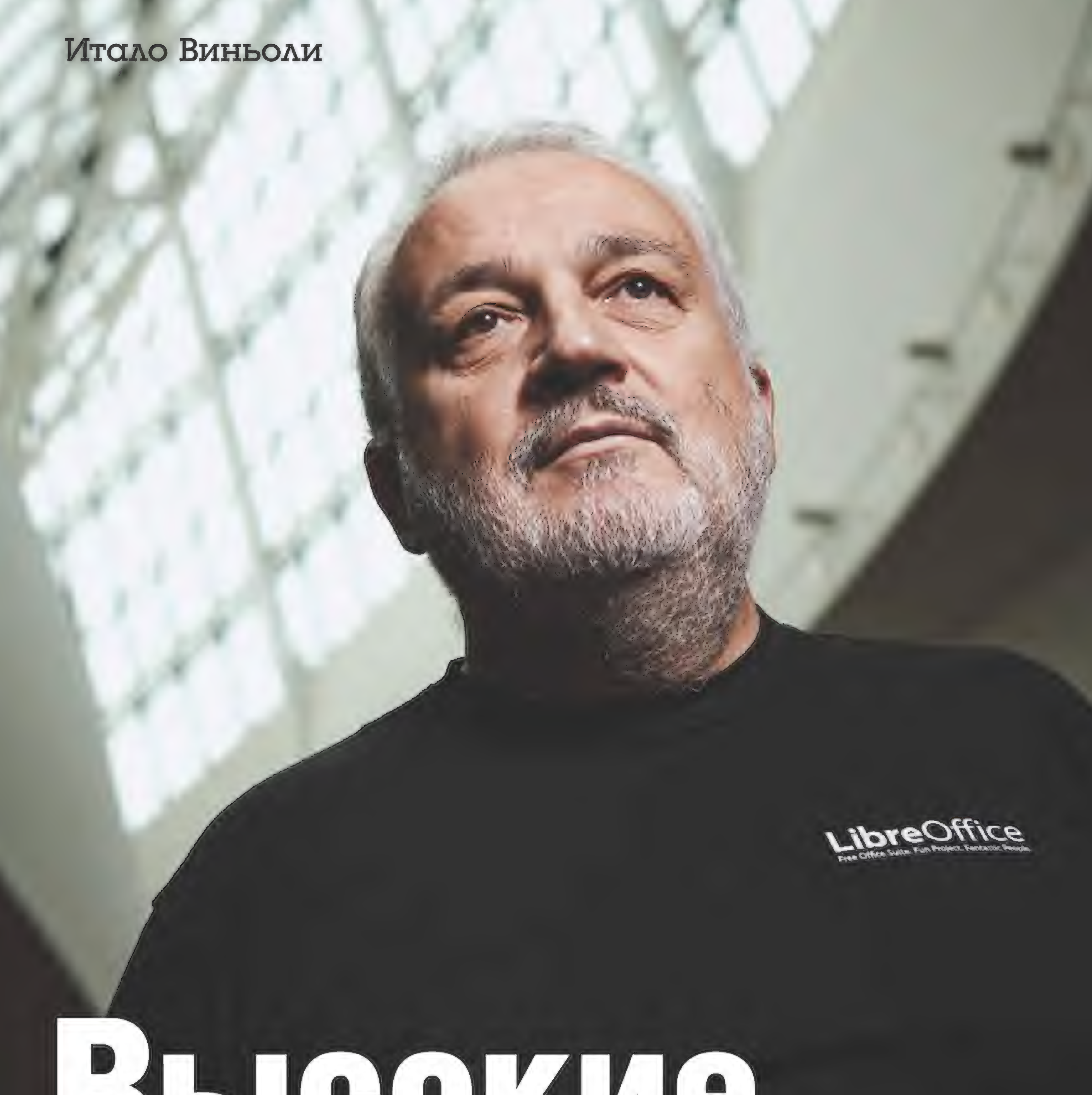
Это намного медленнее, потому что придется читать все файлы, так что попробуйте сузить поиск. Опция **-r** проводит рекурсивный поиск, а **-l** велит **grep** игнорировать двоичные файлы. В большинстве рабочих столов присутствуют также графические команды *find*, но они не могут комбинировать **find** и **grep**, например, так:

```
grep -l "some text" $(find ~ -type f -mtime 0)
```

Это запустит **find** для создания списка файлов, затем передаст его **grep** для осуществления поиска. LXF



Итало Виньоли



Высокие стандарты

Мэтт Хэнсон беседует с **Итало Виньоли** о *LibreOffice*, его отделении от *OpenOffice*, а также о трудностях — и возможностях — становления Open Document Format в качестве стандарта.



Итало Виньоли [Italo Vignoli] — пресс-секретарь The Document Foundation и *LibreOffice*. Мы встретились с ним вскоре после того, как в июле 2014 года британское правительство приняло знаменательное решение о принятии Open Document Format в качестве стандарта.

LXF: Спасибо, что согласились встретиться с нами в такое, должно быть, хлопотное для вас время. Как давно вы работаете в The Document Foundation?

ИВ: Как и большинство сотрудников *LibreOffice*, я начинал в *OpenOffice*. Мой дебют состоялся на второй конференции сообщества OpenOffice.org в Берлине, в 2004 году. Довольно забавное начало, ведь по основной профессии я был — и остаюсь — маркетологом. Где-то в районе 2000 года меня стало раздражать отношение Microsoft к *Office*. Я не технарь, но у меня возникло ощущение, что Microsoft всё больше контролирует мой ПК. Начал искать альтернативы, скачивать бесплатные наборы программ. Потом наткнулся на *OpenOffice* и сказал: «Вот оно! Сразу видно, что сделано по уму», тогда как другие представляли собой всего лишь клоны Microsoft.

Итак, я скачал *OpenOffice*, тогда, наверное, в версии 1.1, где, естественно, были ошибки и проблемы. Затем я начал присматриваться к сообществу и однажды отправил им письмо с текстом: «Ребята, у вас прекрасный продукт, но маркетинг — полный отстой! Возможно, я мог бы вам помочь». Я поехал в Берлин, кое с кем переговорил, и мне сказали: «Да, похоже, вы — тот, кто нам нужен». И так я стал заниматься маркетинговыми стратегиями. Конечно, поскольку я пришёл из другой среды, мне пришлось параллельно формировать свою собственную культуру Open Source.



Поначалу, как и свойственно новичкам, я ошибался, набивал шишки. Но потом, по ходу, я много чему научился и получил море удовольствия, так что затраченное время окупилось сторицей.

LXF: То есть вы действительно ощущали, что к культуре Open Source нужно адаптироваться?

ИВ: Да, и когда зашла речь об отделении [переходе от *OpenOffice* к *LibreOffice*], я естественным образом стал заниматься маркетингом Open Document Foundation.

LXF: А отделение произошло, когда *OpenOffice* перешёл к Oracle?

ИВ: На самом деле, разговоры об этом начались раньше, но, безусловно, Oracle также послужил стимулом. Мы начали давить на Sun, мол: «Послушайте, такая расстановка сил никуда не годится. Роль Sun в системе значительно выше, чем роль сообщества. Мы не желаем выдворить Sun, но хотим расширения прав сообщества». В Sun этому сопротивлялись, но в споре зачастую приходят хорошие идеи относительно того, куда двигаться дальше.

Поэтому на конференции Oracle в конце 2009 года в Орвието, в Италии, я уже был одним из организаторов. И мы ожидали высокопоставленного представителя Oracle — вице-президента по ПО или чего-то в этом роде. А тот парень, что приехал — хороший парень, ничего против него не имею — был просто рядовым сотрудником. Сразу стало ясно, насколько *OpenOffice* «важен» для Oracle: ведь будь это вице-президент по ПО, вывод был бы «ОК, у него есть средства, он может принимать решения», а раз так, стало быть, не будет ни средств, ни решений, проект спустят на тормозах. Это было абсолютно ясно, особенно тем из присутствовавших в Орвието, кто, как и я, имел опыт работы в корпоративной сфере. Поэтому после Орвието, по-моему, сразу после Рождества, мы собрались вновь и решили: нужно отделяться, и немедленно. Иначе проект долго не протянет, и если мы не поспешим, то у нас будут серьёзные проблемы.

LXF: То есть, это было неизбежно? Должно быть, для вас стало ударом то, что Sun не поддерживал *OpenOffice* как полагалось?

ИВ: Да, хотя по-моему, основная проблема — во взаимоотношениях предприятий и сообщества. Возьмите хоть Sun, хоть IBM или Oracle, они сообщество просто не понимают и не принимают. В Sun я постоянно твердил им: «Я знаю, что вам не нравится то, что я говорю, ну так уволъте меня завтра же», а они отвечали: «Да как мы вас уволим, если вы волонтёр!» Я говорил: «Раз вы не можете увольнять волонтёров, вам придётся с ними считаться. Волонтёры никогда не будут так же безропотно принимать решения компании, как сотрудники. Либо вы осознаете, что вам нужно менять свои подходы и взгляды, либо мы никогда не придём к согласию. У меня есть своя работа, свой доход. Волонтёр я — по доброй воле. Не ждите, что я изменю своё мнение. Вы должны найти путь к сотрудничеству с нами. Просто говорить мне, что „такова стратегия", извините, не сработает».

LXF: В наших беседах с руководителями открытых проектов и компаний часто звучит тезис о важности учета мнения сообщества. Причём не только положительного, отрицательное также ценно — возможно, даже более.

ИВ: Именно так, но дело в том, что мне не встречалось ни одного предпринимателя, который действительно понимал бы, что нужно сообществу. Всецело сосредоточенные на компании, они забывают о сообществе. Но нельзя рассчитывать, что сообщество будет полностью согласно с деятельностью компании.

LXF: Правительство Великобритании заявило, что официально принимает Open Document Format (ODF) как стандартный формат правительственных документов. Полагаем, это вас порадовало?

О ФУНДАМЕНТАЛЬНОЙ ПРОБЛЕМЕ

«Корпорации, работающие с Open Source, либо не понимают, либо не любят сообщество.»

ИВ: Более чем. Я от радости в воздухе парил [*смеется*]. С 2006-го я участвовал в переговорах по стандартам между правительством Италии и ЕС. Так что прошло 8 лет, и спустя эти годы, ты словно очнулся ото сна и понял: «О, так это был не сон!» Если правительство что-то принимает, значит, это возымеет действие. И, естественно, становится ясно, что уровень твоей ответственности резко изменился. Теперь, когда это стало реальностью, пора делать следующий шаг, то есть идти в школы, разговаривать с людьми, учить людей составлению стандартных документов, объяснять им, что документ, который они получили в стандартном формате, легко откроется в их приложении, и что они смогут, не мучаясь, так же переслать его обратно. Это как водить машину: вы же делаете это не как вам вздумается, а по правилам. А создавать документ по правилам нам никогда в голову не приходило; но именно об этом и стоит задуматься.

LXF: Ранее вы упомянули о новой ответственности, в связи с тем, что вы теперь работаете над правительственными документами.

ИВ: Совершенно верно. В некотором роде, это мне уже стало привычно, за время сотрудничества с местными органами самоуправления в Италии: начинаешь понимать, что местные власти »

совершенно не разбираются в стандартах. Они не знают, что такое базовая реализация, не знают, что нужно сделать, чтобы проверить совместимость форматов. Так что есть ещё над чем поработать. Не думаю, что британскому правительству удастся на 100% перейти на открытые стандарты за несколько месяцев. Нужно проявить некоторую гибкость, иначе люди станут бояться новых правил, а это самое худшее — ведь страх рождает сопротивление. Скажут: «Не стану я это использовать. Ну и пусть так надо, а мне не нужны правила, создающие мне проблемы»; но мы-то стремимся к обратному. Нам нужно, чтобы люди сказали: «Мы и не думали, что это можно стандартизировать, но здорово, если так».

Я также рад, что правительство Великобритании сделало верный шаг в верном направлении. Это всеобщая ответственность, в том числе и таких компаний, как Microsoft, поскольку они поддерживают ODF в версии 1.2. И, конечно же, в этой связи, очень легко всё испортить, изменив поддержку и нарушив совместимость [с *Microsoft Office*].

Как члены комитета по усовершенствованию стандарта ODF, мы несём дополнительную ответственность, поскольку следующая реализация будет уже не эталонной, но основной для всей страны. Поэтому мы обсуждаем, например, как встроить в ODF шрифты. У нас есть на этот счёт предложение, но мы должны быть открыты и к другим, чтобы это было доступно каждому. *LibreOffice* уже поддерживает встраивание шрифтов, но, конечно же, это не стандарт. Мы должны придумать, как это будет выглядеть в следующей версии ODF, и это не может быть как в *LibreOffice*, это должно быть как в ODF. То есть внедрение шрифтов в ODF должно легко согласоваться с Microsoft, *Apache OpenOffice*, *LibreOffice*, *Calligra*, *AbiWord*, *GNUmeric*. Мы не можем думать только о себе. Только так и создаются стандарты.

LXF: То есть становление ODF в качестве стандарта предполагает много трудностей, но и много возможностей.

ИВ: Да, эти вещи абсолютно соизмеримы. Возможности будут повсюду, ведь, полагаю, с принятием формата правительством компании станут охотнее выбирать *LibreOffice* вместо *Microsoft Office*. Естественно, переход с *Microsoft Office* на *LibreOffice* обойдётся не даром — за лицензию, понятно, платить не придётся, но компаниям, вероятно, потребуется помощь в процессе миграции. И это станет возможностью для тех, кто осуществляет профессиональную поддержку такого рода. Они помогут компаниям перейти со старого шаблона на новый, со старого макроса на новый макрос и т. д.

Сейчас мы разрабатываем — и вскоре запустим — собственную программу сертификации для миграций и обучения. У нас уже есть сертифицированные разработчики; а теперь будут ещё и специалисты, помогающие компаниям при переходе на *LibreOffice* и обучающие сотрудников его использованию.

LXF: Это интересная идея — с сертификацией...

ИВ: В целом, этим будет заниматься TDF (The Document Foundation), но это будет не коммерческая сертификация. Мы будем просить людей покрыть расходы — поскольку, естественно, у проекта они будут — но это не будет обычная проприетарная сертификация, где вы, условно говоря, оплачиваете билет и получаете сертификат. Мы хотим, чтобы люди действительно поняли свободное ПО, поняли сообщество, научились работать с этими программами и прониклись этой философией.

И, наконец, главная возможность, как мне кажется, состоит в продвижении свободного ПО, в том, чтобы помочь людям понять, что использовать его — хорошо и правильно. На самом деле, правильно используя, а значит, поддерживая свободное ПО, вы помогаете развитию всей экосистемы. Содействуете разработке новых функций, исправлению ошибок. В корпоративной среде, по большей части, такое представление всё ещё отсутствует, а значит, это возможность продемонстрировать им, что это перспективно, поскольку лично я убеждён, что путь развития Open Source — это и есть дальнейший путь развития ПО. Здесь шире сообщество разработчиков. Прошли те дни, когда компания могла собрать всех разработчиков проекта в одной комнате. Конечно же, Microsoft эта идея не по душе, но все мы знаем, что они начали разрабатывать ПО 30 лет назад и, вероятно, никогда не думали, что это можно делать как-то иначе. Open Source показывает им, что есть и другие варианты, на которые уже пора обратить внимание.

LXF: Ну, в этом году они поприсутствовали на OSCON (Open Source Convention).

ИВ: Скажем так: они проявляют любопытство. Проблема в том, что я хорошо знаю людей,



которые сотрудничают с Open Source. У них правильное мышление, но им, к сожалению, приходится иметь дело с компанией, которая в основном защищает старые взгляды.

LXF: Полагаю, их это весьма пугает? Microsoft уже как-то отреагировал на решение британского правительства принять стандарт ODF?

ИВ: Не особенно. Думаю, сейчас ещё не время задавать им вопросы на этот счёт, ведь на протяжении 20 лет они через свои офисные форматы правили миром. И впервые кто-то осмелился дать им оплеуху, сказав: «Ваши форматы неправильные». Думаю, справедливо будет подождать, чтобы они поняли, в чем проблема их метода. Ведь когда Мюнхен успешно переходит на Linux, когда французское правительство выбирает *LibreOffice*, а британское правительство — ODF, и многие регионы Италии и Германии выступают за свободное ПО, по-моему, пора понять, что нельзя больше занимать позицию в духе: «Мы защищаемся, лоббируя свои интересы, свои устои». Либо вы движетесь в сторону совместимости, либо мир будет двигаться без вас. Я полагаю, они не ожидали столь брутального, скажем так, решения от британского правительства. Но им придётся с этим считаться.

LXF: Подстегнуло ли это вашу веру в то, что другие правительства последуют их примеру?

ИВ: Само собой. Думаю, это как с лавиной: сначала одна снежинка, затем другая, и пошло-поехало. Думаю, дело было за тем, какое правительство осмелится сделать первый шаг — и этот шаг был сделан. Другие должны увидеть это и решить: «Есть такая возможность, британское правительство её приняло, у них имелась мотивация, они проявили усердие, провели кучу исследований». Не просто же так, кто-то с утра проснулся и сказал: «Давайте примем ODF!» Это хорошо продуманный процесс. Так что, думаю, пример Великобритании будет всерьёз рассмотрен в Европе и, возможно, в Южной Америке, поскольку именно в этих регионах свободное ПО распространено более всего. Не говоря уже о том, что в Италии есть люди, активно призывающие правительство принять решение.

LXF: Так почему же, по-вашему, организации и правительства не решаются переходить на форматы Open Document?

ИВ: Думаю, в некотором роде так сложилось, с учётом того, что Microsoft был, во многом, столь успешен, создав первый настоящий офисный набор. Они первыми дали то, что пользователям требовалось. Поэтому их и приняли, за ними пошли. А Microsoft, естественно, начал закрывать продукт раньше, чем это стало заметно. И со временем мы обросли этими файлами **.doc** и **.docx**, и прочим, вот правительства за них и держатся.

Но это уже история, а, как показывает практика, историю человечество способно изменять. Пусть и со временем. На самом деле, сложившаяся ситуация замедлила принятие решений и добавила правительствам сомнений. Полагаю, имеет место и скрытый вопрос: «А что, если?» Что если я перейду на формат ODF, когда весь остальной мир использует — не как стандарт, но использует — другой формат? Если вы уверены, что ODF лучше, тогда это решение — только вопрос времени. Но протянув, мы лишь увеличим количество проблем, связанных с его принятием. Естественно, придётся конвертировать море документов. OK, это объяснимо. Но если прождать ещё 10 лет, документов станет и того больше. Так что, я думаю, существующее сопротивление объяснимо, но мы очень рады, что правительство Великобритании выбрало революционный, а не эволюционный путь. И мы готовы помочь сделать это реальностью.

LXF: Приятно видеть, что британское правительство оказалось в чём-то прогрессивным. Ведь ещё ранее в этом году они платили Microsoft за продление поддержки XP.

ИВ: На самом деле, это было весьма неожиданно. Ещё накануне вечером я разговаривал с подругой, работающей в правительстве, и она говорила: «Ты знаешь, мы ждём принятия решения, и я хотела бы сказать тебе что-то определённое, но пока у меня нет такой информации». Я пошёл спать, а на следующее утро прочитал об этом пресс-релиз — так что это был большой сюрприз! Только вечером мы это обсуждали, и моя подруга сказала, что «придётся подождать ещё 15 дней», а не 12 часов!

LXF: Приятная оперативность.

ИВ: Конечно! Но она была не в курсе. Она сказала: «Решение уже принято, но теперь идёт политический процесс для оглашения. Мы, чиновники, этим не занимаемся — это дело политиков».

LXF: А когда дело касается политики и политиков...

ИВ: Да уж, тут никогда не знаешь, как повернётся. Они могли это спокойно отложить на какой-нибудь другой день, когда у правительства не будет других заявлений.

LXF: Возвращаясь к тому, о чём вы говорили ранее, похоже, что преимущество Microsoft — в большом количестве людей, считающих, что их форматы и их офис — неизбежность. Однако во все больше дистрибутивов по умолчанию пробирается *LibreOffice*. Это вас радует?

ИВ: Безусловно, но я считаю, что это следствие того, как появился проект. Поскольку он общественный, естественно, что Linux-сообщества его используют. Есть и другие чисто технические причины, скажем, временной цикл релизов, выбранный нами исключительно в угоду дистрибутивам Linux. У Windows тоже есть цикл релизов, но никто вам не скажет, когда следующий. На слова Microsoft «Мы его выпустим тогда-то» полагаться нельзя. Так что подстраивать свою стратегию под Microsoft бессмысленно.



Растёт популярность Mac OS, но она закрытая, и мы не видим проблем с тем, что наш цикл обновления зависит не от нее, а от Linux, и выходим дважды в год, в июле и в январе. Если сопоставить с крупнейшими проектами — Ubuntu выходит в апреле и в октябре, так что мы даём им достаточно времени включить наш основной релиз в их основной релиз. Fedora обновляется реже, но по такому же принципу. Поэтому я считаю, что наша стратегия релизов завязана на Linux.

О РЕАКЦИИ НА РЕШЕНИЕ ВЕЛИКОБРИТАНИИ

«Думаю, это как с лавиной: сначала одна снежинка, затем другая, и поехало...»

И конечно, это принесло свои плоды. Равно как и то, что члены нашего сообщества являются также и членами сообществ Linux: например, в Ubuntu это Бьёрн Михаэльсен [Björn Michaelsen] [из Canonical] — один из членов правления и ответственный за подготовку пакета *LibreOffice* для Ubuntu. Рене Энгельхард [Rene Engelhard] также занимается *LibreOffice* для Debian и является его членом. То же самое и с Fedora. Думаю, наш роман с Linux можно объяснить и тем, что у нас общие корни.

LXF: Этот роман завязался естественным образом, но были ли у *LibreOffice* связи с другими дистрибутивами?

ИВ: Да, конечно. Большинство наших разработчиков — это пользователи Linux, работающие с самыми разными проектами. Это и Майкл Микс [Michael Meeks] из Gnome, Каолан Макнамара [Caolán McNamara] из Red Hat, также и с Fedora, также и с CentOS, так что всё очень органично. LXF

Криптография старая и новая

Присоединяйтесь к путешествию Джонни Бидвелла по миру создания и взлома кода, миру тайн и интриг...

Сколько существуют на свете истории, столько же существуют и секреты – слова, не высказанные из тактических соображений или из страха расправы. Часто секреты приходится передавать вдаль, и крайне важно, чтобы они оставались скрытыми на всем пути следования. Именно так произошло, когда планы Ксеркса по захвату Спарты были разрушены Демаратом (изгнанником, живущим в Персии, который отослал предупреждение в Спарту на с виду пустой восковой табличке). И то же касается информации по вашей кредитной карте, которую вы пересылаете при оплате купленных вами гаджетов, продуктов или носков.

Вероятно, большинство людей знакомы с подстановочными шифрами, в которых одна буква заменяется другой. Самый известный из них — шифр Цезаря: каждая буква заменяется другой, расположенной от нее на определенном удалении, а когда алфавит заканчивается, все повторяется сначала. Говорят, что этим методом пользовался Юлий Цезарь, заменяя A на D, B на E и так далее; в конце A заменяла X. А его племянник Август предпочитал сдвиг на одну букву, где A заменялась на B, B на C, но циклического перехода не было, так что Z просто заменялась на символ AA.

Кама Сутра, помимо прочих интересных вещей, упоминает также искусство млекхита-викальпа (тайнописи). Там подробно описывается подстановочный шифр, в котором буквы сгруппированы по парам и взаимозаменяемы по фиксированной случайной схеме, чтобы любовники могли «скрывать подробности своего романа». Еще более древняя система подстановки — Атбаш, она встречается с древних (около 500 г. до н.э.) иудейских текстах. Здесь первая буква алфавита, алеф, заменяется на последнюю, тав; вторая, бет, на вторую с конца... и так далее, что весьма эффективно меняет буквы алфавита местами. Эквивалент на латинице — заменить A на Z, B на Y, ну и так далее. Система ROT13 (шифр Цезаря со сдвигом в 13 букв) по-прежнему используется на некоторых сайтах и в новостных группах, чтобы скрыть спойлеры, кульминационные моменты или бранные слова.

> «Кама Сутра, помимо прочих интересных вещей, упоминает искусство тайнописи.»

Все эти моноалфавитные подстановочные шифры (MSC) ни в коей мере не безопасны по стандартам современной криптографии, но в свое время были вполне эффективными — разбойники с большой дороги во времена Цезаря вряд ли отличались грамотностью, в отличие от искушенных мастеров слова современного Интернета. Но все же эти шифры содержат в себе зародыш идеи современного криптографического ключа. Будь то длина буквенного сдвига в шрифте Цезаря, диаметр скиталы или пары, используемые в Кама Сутре (нет, мы не те пары имели в виду), знание метода шифрования и его ключа позволяет дешифровать сообщение.

Для шифра Цезаря у нас есть 26 возможных ключей (включая сюда тривиальный нулевой сдвиг), тогда как ROT13 и Атбаш по сути являются одноключевыми системами. У шифра Кама Сутры довольно большой набор ключей — около 8 триллионов (8 с 12-ю нулями) уникальных способов подбора алфавитных пар. В обычном MSC — потрясающее количество возможных комбинаций (26-факториал — около 4 с 26-ю нулями — или чуть больше 88 бит в современном двоичном исчислении), однако размер — не главное... Арабский ученый Аль-Кинди в манускрипте IX века под названием «О дешифрации криптографических сообщений» приводит первое описание взлома MSC с помощью частотного анализа, который использует тот факт, что в «среднем» сообщении одни буквы используются чаще других.

Например, в английском языке буква "e" имеет относительную частоту использования около 13%, за ней следует буква "t" — 9%, и так далее. Именно поэтому счет в игре Scrabble [русский аналог — игра «Эрудит», — *прим. ред.*] ведется таким образом, а не иначе: чем чаще употребляется буква, тем меньше за нее очков. В других языках — другие буквы и частотные характеристики, однако принцип остается тем же: замените самую часто встречающуюся букву в шифре на самую часто встречающуюся букву алфавита, затем примените то же правило к следующей по частотности букве, и продолжайте в том же духе, пока не заполните все пробелы. В исходном сообщении не обязательно будет та же частота употребления букв, что и в языке, но при условии, что оно достаточно длинное, оно, по крайней мере, будет достаточно близко к статистике, чтобы дешифровка стала возможной с небольшими догадками.

Раскрытие в 1586 году заговора Бабингтона (целью которого было убийство королевы Елизаветы I) привело к казни шотландской королевы Марии Стюарт и ее сообщников; а все потому, что их переписка была раскрыта знаменитым дешифровщиком Томасом Фелиппесом [Thomas Phelippes]. Переписка между Марией Стюарт

Без паники, полковник



LINUS
XFGAD
AGFGA
AGDXG
DGFFD
FA

ILNSU
FXGDA
GAFAG
GADGX
GDFDF
AF

Этот триптих демонстрирует еще один пример из времен Первой мировой: шифр ADFGX (эти буквы были выбраны потому, что в азбуке Морзе они разные).

Первая часть — это фракционирующий ключ: он кодирует каждую букву латинского алфавита (кроме буквы z, которая не одобрена руководством по стилю **LXF**) в биграмму, и наше сообщение «паника ядра» шифруется, превращаясь в XF GA DA GF GA AG DX GD GF FD FA (пробелы игнорируются). Во второй части мы помещаем это сообщение в сетку под вторым ключевым словом, LINUS, которое является нашим ключом перестановки. На практике используются более длинные ключи перестановки, и оба ключа изменятся в соответствии с ежедневной шифровальной книгой. Мы изменили порядок столбцов, поставив второй ключ в алфавитном порядке и затем считывая шифр по столбцам. И, таким образом, наше зашифрованное сообщение выглядит следующим образом: FGGGA XAADF GFDF DAGD AGXF.

и Бабингтоном была зашифрована методом подстановки, использовавшим в основном буквы греческого алфавита, и Фелиппес сумел сфальсифицировать дополнение к одному из писем Марии Стюарт, спрашивавшее имена заговорщиков. После того, как их обвинили, все они были обезглавлены.

Значимой вехой в истории криптографии стало изобретение в 1553 году так называемого шифра Виженера [Vigenère cipher]. На самом деле его сработал криптолог Джован-Баттиста Беллазо [Giovan Battista Bellaso] на основе идей Тритемиуса [Trithemius] и Альберти [Alberti]. Виженер действительно опубликовал (более сильный) шифр на автоключе в 1586 году, но история ошибочно приписала более ранний шифр ему. Этот шифр является полиалфавитным подстановочным шифром, использующим ключевые слова для перехода на другой алфавит шифрования после каждой буквы. Каждая буква зашифрована шифром Цезаря со сдвигом, определенным соответствующей буквой ключевого слова. Это (при условии, что в ключевом слове более одной уникальной буквы) не позволяет провести частотный анализ. Шифр считался настолько сильным, что был признан *le chiffre indéchiffrable* — не поддающимся дешифровке, и он оставался таковым вплоть до трудов Бэббиджа [Babbage] и Касиски [Kasiski] в середине XIX века. Их усилия были направлены на установление длины ключа: если она известна, можно разделить шифрованный текст на множество отрезков; каждый отрезок подвергался дешифровке с помощью разных цезарианских ключей, с которыми легко справлялся частотный анализ.

Позднее этот шифр был дополнен еще и буквой V, чтобы создать изобретательно названный шифр ADFGVX. В 1918 году французский криптоаналитик Жорж Пенвен [Georges Painvin] чудесным образом умудрился прочесть сообщение, зашифрованное с помощью ADFGVX, в котором сообщалось о планах немецкой армии напасть на Париж. За время работы по дешифровке Пенвен похудел на 15 кг.

Возможно, кому-то интересно, можно ли в принципе создать шифр, не поддающийся взлому, и кто-то будет сильно удивлен, узнав, что такой шифр уже есть. А то, что он запатентован в 1917 году, может кого-то столь сильно ошеломить, что причинит вред здоровью; и тем не менее, таков факт. Тот, кто за все это отвечает (ну, по крайней мере, за патент) — Гилберт Вернам [Gilbert Vernam], а его изобретение известно как «одноразовый код [One Time Pad]». Хитрость в том, что материала для ключа столько же, сколько текста, что материал для ключа исключительно случаен и совершенно секретен, и что ни одна часть материала для ключа не используется более одного раза. На практике, однако, система Вернама по большей части бессмысленна. Создать действительно случайный материал сложно, так же, как и распространить его большое количество в условиях строгой секретности и обеспечить уничтожение после использования.

Загадочная математика и Enigma

Криптография военного времени в основном полагалась на шифровальные книги, содержавшие ежедневные ключи, но у тех была скверная привычка периодически попадать в руки врага. Когда происходила подобная утечка и новости о ней достигали штаба, генералитет сталкивался с колоссальной логистической проблемой предупреждения персонала об утечке и создания и распространения нового материала ключей. Нередко дальние военно-морские миссии не получали их и продолжали отправлять сообщения, используя старые ключи. Иногда эти сообщения перехватывали и получали информацию о новом ключе. Во время Первой мировой войны дешифровка телеграммы Циммермана (в которой Мексике предлагалось стать союзником Германии) стала решающим фактором вступления Америки в войну.

К началу Второй мировой войны немцы обновили и модернизировали серию Enigma, что стало солидным криптографическим вызовом для Блетчли-парка. Польские исследователи взломали изначальную структуру еще в 1932 году и перед самым началом войны поделились своими данными с Британией. Алан Тьюринг [Alan Turing] создал устройство Bombe, которое к 1940 году прекрасно справлялось с задачей дешифровки сообщений нацистов.

Устройство Enigma, несмотря на огромное количество валов, наборных панелей и штекерных настроек, все же имело слабость, заключавшуюся в том, что буква никогда не шифровалась самой собой. Это солидно сокращало работу Bombe и вычислителей (обычно это были женщины, внимательные к деталям и умевшие хорошо разгадывать кроссворды). Когда в Enigma печаталась буква, алфавит шифрования изменялся роторным механизмом таким образом, который не отличался от шифра Виженера. Были и другие уровни шифрования, однако многие из них были постоянными настройками, которые стали ненужными после того, как устройства Enigma были захвачены. К концу войны по всей Англии использовалось »

около 200 Bombes. Американцам, у которых было куда меньше проблем с материальной базой, удалось создать 125 более быстрых Bombes, а Союзники могли пожинать плоды работы этих монстров через (зашифрованные) линии.

Независимо от гения Тьюринга, большая часть трафика Enigma была дешифрована благодаря плохо организованной операционной безопасности. Ключи сообщений могли бы изменяться с каждой передачей данных, но этого не происходило; или, если изменения все же были, они были незначительными, и их можно было легко просчитать. Цифры часто писались буквами, так что обычной техникой был поиск совпадений, которые можно было расшифровать, как eins — *нем.* «один». Будь разрешены нумералы, эта техника бы не работала.

В 1970-х две разработки вывели игру в криптографию в компьютерный век. Первой из этих разработок был Стандарт шифрования данных [Data Encryption Standard] — блочный шифр, основанный на работе Хорста Файстеля [Horst Feistel] в IBM. До своей стандартизации он был чуть изменен по приказу АНБ. Поскольку никто не объяснил причины этих изменений, возникли подозрения об утечке. Двумя десятилетиями позднее выяснилось, что все было как раз наоборот: S-блоки оригинального шифра были уязвимыми для техники, именуемой «дифференциальным криптоанализом», которая на тот момент (поскольку криптография считалась военной технологией) была засекречена. Изменения АНБ сделали шифр более устойчивым к этой технике, хотя они также рекомендовали использовать более короткий, 48-битный размер ключа вместо 64-битного. Поскольку это был первый общественно доступный шифр, DES стал объектом очень пристального изучения и во многом подстегнул серьезное академическое изучение криптографии.

Хотя тысячи страниц журнальных статей по данному предмету приводят самые разные теоретические атаки на DES, его самым серьезным слабым пунктом является короткий размер ключа. IBM и АНБ в конечном итоге согласились на номинально 64-битный ключ, но восемь из этих 64 битов были избыточными битами контрольной суммы. На момент появления этого было, вероятно, вполне достаточно, но в начале 1990-х появились устройства, способные взломать ключ за несколько часов. В 1997 году проект, которых охватил весь Интернет, впервые успешно справился со взломом ключа DES. В 1998 году Electronic Frontier Foundation создал устройство (за что им по-царски заплатили $250000), которое взломало ключ чуть менее чем за два дня.

Среди прочих атак на DES стоит упомянуть «линейный криптоанализ» Мацуи. Атака включает создание приблизительных значений частей шифра посредством нахождения линейных выражений по модулю 2, имеющих вероятности, значительно отличающиеся от 0,5. Собрав большое количество пар (2^{43}) «обычный текст — шифрованный текст», можно выяснить достаточное число битов ключа, чтобы остаток легко взламывался. Линейные выражения весьма быстро отыскиваются благодаря преобразованию Уолша–Адамара [Walsh–Hadamard], и современные шифры стараются включать в себя сугубо нелинейные компоненты, снижающие риск таких атак. В некотором смысле работу Мацуи можно считать посвященной анализу обычной частотности употребления букв, которая вместо языка использует характеристики шифра, а вместо символов — единицы и нули.

Выход на публику

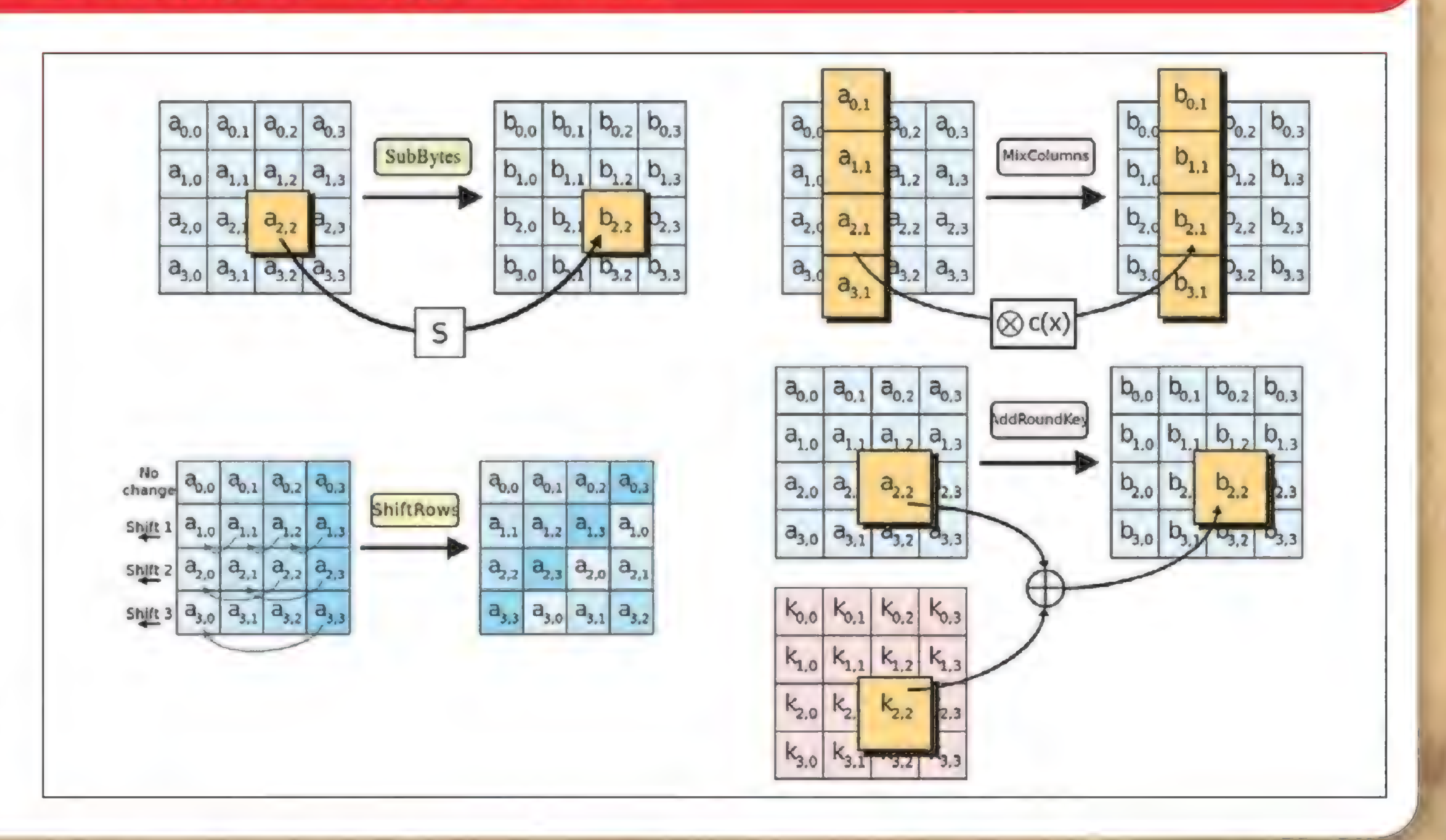
Еще один дар 1970-х — криптография публичного ключа [Public Key Cryptography]. Она наконец-то решила проблему безопасности общения, обойдясь без личной встречи для установления условий сохранения секретности. Этот метод называется обменом ключей Диффи–Хеллмана [Diffie–Hellman], по имени его открывателей. Он использует хиральную математику конечных полей, в которой легко потенцировать элемент (то есть возвести число в степень), но очень сложно провести обратный процесс, известный как дискретное логарифмирование. Так что возведение в степень на этом поле является примером «необратимой функции». Иллюстрация (внизу следующей страницы) показывает пример обмена между Алисой и Бобом, которые весьма недурно изучили литературу по криптографии. Их общий секрет — $s=g^{ab}$ — могут вычислить и Алиса, и Боб. Наблюдателю — скажем, Оскару — видны публичные ключи **A** и **B** и параметры обмена **g** и **p**, однако ему не вычислить сам секрет, не разузнав также один из приватных ключей, **a** или **b**.

Установленный однажды таким образом, общий секрет **s** может использоваться как краткосрочный ключ шифрования для симметричного шифра типа DES. Секретные ключи **a** и **b** в этот момент можно уничтожить, что обеспечит так называемую идеальную прямую секретность, но должная инфраструктура публичного ключа требует, чтобы приватный и публичный ключи оставались по большей части неизменными. Далее, публичные ключи следует как можно шире прорекламировать, для снижения шансов того, что человек-посредник, скажем, некий Мэллори, выдал бы себя за одну из сторон с фальшивым публичным ключом: обмен ключами обеспечивает конфиденциальность, но сам по себе не гарантирует подлинности. Чтобы обеспечить подлинность, нужно точно знать, какой публичный ключ кому принадлежит. Для этого нужна доверенная третья сторона, известная как центр сертификации [Certificate Authority] (CA), которая будет служить справочником владельцев ключей.

Поскольку криптография публичного ключа сильно отличается от его приватного собрата, можно использовать всякие математические

Расширенный стандарт шифрования

AES [Advanced Encryption Standard] появился в качестве замены DES в 2001 году. До настоящего времени он противостоял всем попыткам криптоаналитики найти в нем слабости. Одной из причин его выбора была его относительно простая структура. Есть четыре основных слоя, которые повторяются через определенные промежутки. При наличии хотя бы капли воображения можно увидеть здесь отголоски шифра ADFGX на стадии ShiftRows. Стадия SubBytes — единственная нелинейная часть шифра. Обычно линейные операции выполняются намного быстрее, но без нелинейной стадии шифр будет очень легко взламываться методом Мацуи.



Разработка современных принципов

За последние 150 лет было разработано несколько ключевых принципов, которые (с небольшими изменениями по причине появления новых технологий) по-прежнему дают неплохое представление о том, в чем же суть криптографии. Первый — принцип Керкхоффса [Kerckhoffs]: само по себе знание метода шифрования не должно считаться угрозой безопасности сообщения. Пока ключ не скомпрометирован, это знание не поможет. Он противоположен идее безопасности в силу неведения, которая, хотя и может показаться интуитивно разумной, считается сегодня дурным тоном. Система защиты от копирования CSS, которая используется на DVD, была взломана в 1999 году после того, как обратный инжиниринг программ *Xing* выявил ключ воспроизведения и его алгоритм (оказавшийся трагически слабым). Подобным же образом в 2006 году был взломан механизм KeeLoq для дистанционного управления замком автомобиля, когда часть его конструкции стала известна в результате утечки.

Клода Шеннона [Claude Shannon] часто называют основоположником теории информации. В 1949 году он ввел для шифров понятия смешения и рассеяния. Смешение утверждает, что связи между шифротекстом, обычным текстом и ключом должны быть как можно сложнее. С точки зрения современных блочных шифров это должно означать, что каждый бит результата зависит нелинейным образом от нескольких битов ключа и ввода. Рассеяние опирается на идею, что изменение одного бита ключа или ввода должно радикально менять результат. Критерий идеального рассеяния — жесткий лавинный эффект: каждый бит результата при изменении одного бита ключа или ввода должен изменяться с вероятностью 0,5.

хитрости, чтобы значительно сузить область поиска по сравнению с лобовой атакой. И по этой причине все классические алгоритмы публичных ключей имеют куда более длинные ключи. Например, алгоритм AES считается безопасным при наличии 128-битного ключа; однако люди уже озабочены небезопасностью 1024-битных ключей RSA. Новомодная криптография на основе эллиптических кривых, опять же на базе дискретных логарифмов, но в более абстрактном алгебраическом пространстве, предлагает ключи покороче и все же почти удвоенный параметр безопасности.

Безопасность всех этих систем публичных ключей полагается на предполагаемую невозможность раскрыть факторизующие целые числа и задачу дискретного логарифма. Хотя математики очень тщательно изучили эти проблемы и предложили весьма недурные варианты увеличения скорости процесса, обе они по-прежнему требуют немалого времени на решение и продолжают считаться безопасными — по крайней мере, на обычном оборудовании.

До 1992 года криптографические программы в США были засекречены как часть военных программ, и даже после этого года сохраняют ограничения по экспорту: без лицензии не разрешается экспорт любых программ, использующих длину ключа более 40 бит. Это привело к длительному следствию над основателем PGP Полом Циммерманом [Paul Zimmerman], которое закончилось ничем. Циммерман предложил весьма нетривиальные способы обойти ограничения, включая публикацию исходного кода в виде книги, защищенной Первой Поправкой [поправка к Конституции США о невмешательстве государства в вопросы религий, — *прим. ред.*]. Компанию Netscape принудили выпустить убогую “International Edition”, допускающую только 40-битные ключи SSL, против 128-битной американской редакции.

Вы Шор?

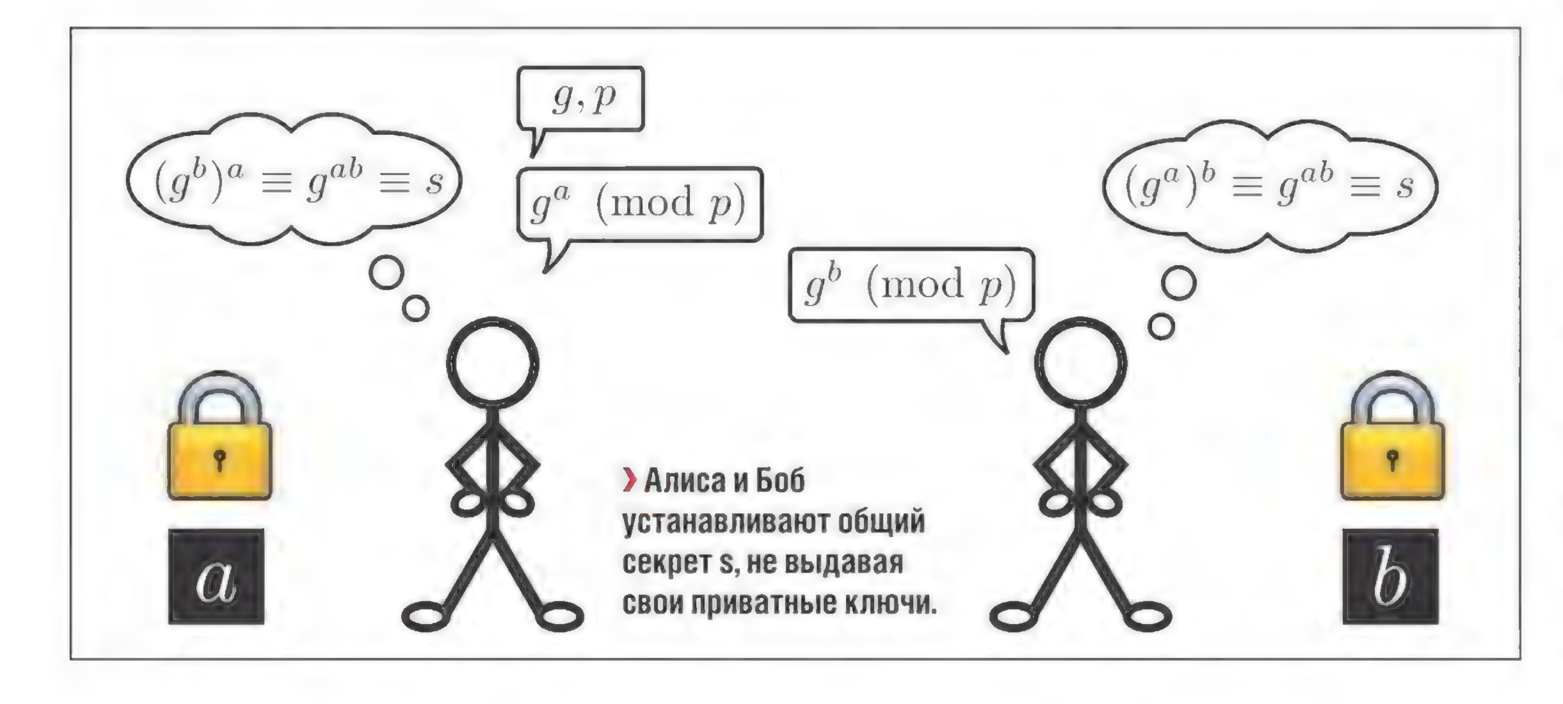
В 1994 году Питер Шор [Peter Shor] объявил о создании алгоритма, с которым можно работать на квантовом компьютере и который позволяет (помимо всего прочего) факторизовать целые числа и вычислять дискретные логарифмы намного быстрее, чем на классическом компьютере. Хотя никто пока что не преуспел в создании нужного типа квантового компьютера, имеются достаточные предпосылки для разрастания области исследований, известной как постквантовая криптография.

Вероятно, более практической является проблема создания безопасных ключей. Она предполагает возможность производить достаточно случайный поток битов, с которыми компьютеры исключительно плохо справляются. В Linux у нас есть узлы **/dev/random** и **/dev/urandom** (попробуйте-ка запустить в них команду **cat**), которые берут энтропию, собранную (помимо других источников) с ввода с клавиатуры и мыши, чтобы пополнить псевдослучайный генератор чисел (PRNG). Вот почему очень мудро совершать бессмысленные движения мышью и лупить по клавиатуре при запуске, например, команды **ssh-keygen**.

Очень ранняя версия *Netscape* содержала слабый PRNG, который инициировался временем дня и id процесса. Поскольку атакующий вполне может выдвинуть разумные предположения об этих переменных, то якобы случайно генерируемые ключи SSL подвержены взлому. В 2008 году был шквал паники у системных администраторов, когда выяснилось, что OpenSSL генерировал слабые ключи — и занимался этим года два. А недавно Эд Сноуден [Ed Snowden] сообщил, что АНБ платили RSA security за использование по умолчанию в своем ПО генератора под названием Dual EC DRBG. Есть подозрение, что постоянные, которыми АНБ рекомендует инициализировать этот генератор, придуманы таким образом, чтобы создать в алгоритме лазейку.

Помимо шифров, есть такая важная концепция, как хэш-функция. Она кодирует вводимую информацию в выводимые данные определенной длины (и если ввод длиннее вывода, то могут быть коллизии) необратимым образом. Хэшированные пароли в Linux хранятся в **/etc/shadow**. Изначально использовался алгоритм хэширования MD5, но ныне стандартом становится SHA-512. Мы нередко слышим, что хакеры взломали базы данных, содержащие хэшированные пароли. Если вы владеете большой базой данных, популярная программа взлома паролей *John the Ripper* в состоянии за считанные минуты вытащить все слабые пароли. Для исследовательских целей мы запустили ее на вполне реальной базе данных (на несколько тысяч пользователей), и буквально за пару-тройку часов выудили 2500 паролей. Другие инструменты, такие, как *oclHashcat*, могут еще и привлечь мощности GPU, так что безопасность базы данных очень важна — как и смена вашего пароля, если он оказался скомпрометирован.

Итак, мы наблюдаем большие изменения в способах шифрования наших секретов, однако важно видеть, как нас вдохновляет прошлое. Увы, мы совершаем все те же ошибки — если в системе безопасности происходит взлом, то, вероятнее всего, из-за чьей-то неосторожности, а не из-за слабости шифра. Плохо настроенные серверы, фишинговые атаки, злонамеренные или ленивые операторы являют собой куда более серьезную проблему. LXF



› **Алиса и Боб устанавливают общий секрет s, не выдавая свои приватные ключи.**

Linux пошел в школу

В сентябре 2014 Англия стала первой в мире страной, где обучение программированию будет вестись с младших классов. Шивон Рэмси делает шаг в это светлое будущее.

По всей стране школы бьются над поиском методов обучения визуальным и текстовым языкам программирования, которые позволят детям создавать и отлаживать свои собственные программы. Разбираются с сетевой терминологией, установкой разных операционных систем, конструированием статических и динамических сайтов с исполнением скриптов на сервере. Учатся использовать открытые библиотеки для управления компьютером и преподавания таких языков, как Python, JavaScript, Ruby, C++ и C.

И всё это из-за нового учебного плана по информатике — см. http://bit.ly/LXFnaace. Учителей, мало знакомых с компьютерами, «железом», Linux или азами программирования, это может напугать. Как вариант, школы могут преподавать CS "unplugged" [Computer Science Unplugged — способ обучения программированию без компьютеров, — *прим. пер.*], внедряя практические задания, позволяющие детям понять принципы компьютерных наук. Однако информатика по сути своей дисциплина прикладная и требует непосредственной практики в построении и программировании компьютерных систем, чтобы дети могли понять, как они работают, и научиться обнаруживать и исправлять ошибки.

Новый учебный план открывает IT-специалистам и энтузиастам Open Source прекрасную возможность наладить контакт со школами, поделиться опытом и вдохновить детей мыслить в логике программирования и использовать технологию как средство для получения знаний.



› За плечами у Шивон — более 18 лет работы с Open Source.

Компьютерное мышление

Недавно мы общались с приятелем, сыну которого на пять лет подарили робота. Поиграв с ним всего несколько минут, посмотрев, как он ходит вверх и вниз, бегает, прыгает, крутится и поворачивается, он спросил: «Пап, это всё, что он умеет?»

Совсем другое дело — дать детям шанс создать какую-нибудь простенькую программу самостоятельно с помощью языка drag-and-drop. Тогда с помощью компьютера и силы своего воображения они смогут найти решение простой проблемы. Научатся задавать вопросы, понимать мир вокруг себя и формировать его. И получат удовольствие от того, что их решение работает, и что они внесли свою лепту! Мой опыт наставничества и обучения детей и молодёжи в сфере прикладных программных проектов подтверждает, что такие методы очень эффективны.



› Проекты Raspberry Pi — эффективный путь обучения программированию.

В основе этого учебного плана лежит стремление обучить молодых людей компьютерному мышлению. Этот термин придумала программист Жаннетта Винг [Jeannette Wing], для определения особого типа мышления, необходимого для построения задач, решаемых компьютером. Люди, которые им обладают, черпают информацию вокруг себя и переводят её в данные для компьютерной обработки. Они упрощают реальную ситуацию, отбрасывая лишние детали, несущественные для решения насущной проблемы (абстракция). Разбивают сложный процесс на более мелкие части или подпроблемы (декомпозиция). Выявляют естественные и вычислительные схемы и применяют логику для обобщений, которые они тестируют, внедряют и улучшают с помощью обратной связи.

Жаннетта Винг считает вычисления «автоматизацией наших абстракций». Большинство компьютерных проектов, как виртуальных, так и прикладных, построены по принципу упрощения реальных явлений, но многие с помощью компьютеров находят решения существующих проблем. В настоящее время ведётся активный поиск ответов на вопрос, как нам использовать тот объём информации, который сегодня доступен обо всех сферах нашей жизни, чтобы улучшить её качество, от создания автоматических систем отопления до конструирования управляемых наружных каркасов, которые помогут парализованным людям самостоятельно ходить, вызывающих немало споров использования алгоритмов для охраны правопорядка и разработки индивидуальных лекарств.

Профессиональная поддержка

Развитие компьютерного мышления стал основой для создания учебного плана, поскольку от этого зависит мировосприятие молодого поколения. В образовательной программе по информатике описано, чему именно должны обучать детей на каждом из ключевых этапов [Key Stage] (см. врезку внизу). В ней три области — информатика, технология и компьютерная грамотность; но главную цель и, возможно, самую большую трудность для школ представляет собой введение в компьютерную науку.

> «Компьютерное мышление в учебном плане приоритетно, поскольку фундаментально.»

Это очень сложный материал; кто же поможет с ним учителям? Naace, ассоциация ИКТ — это сообщество преподавателей, технических специалистов и управленцев, объединённых представлением о роли технологии в развитии образования, тех, кто работает в школах и с школами. Они позиционируют себя как независимое сообщество с 30-летним стажем, которое, не в угоду кому-то, активно представляет интересы всех.

Naace всегда интересовалась использованием технологий в школах, в особенности их влиянием на процесс обучения, преподавания и организации управления. В этой связи, именно члены Naace впервые отметили, что использование компьютеров может помешать хорошему учителю, но уж точно не поможет плохому.

И это абсолютно справедливо в отношении программирования как нового предмета, и особенно той его части, которая касается информатики. Чтобы этого избежать, Naace организует обмен опытом, как между школами, так и с привлечением внешних специалистов, ведь в составе Naace — передовые разработчики, готовые бесплатно делиться своими знаниями посредством информационных рассылок и сайтов.

Для прямого обучения существует Picademy, предлагающая двухдневный курс для учителей всех специальностей — не только тех, кто преподаёт Информатику, а кто хочет задействовать компьютер и электронику в своей работе. Цель курса в формировании навыков и уверенности, а также обмене положительным опытом. В первый день учителя посещают мастер-классы по Sonic Pi, созданию прикладных проектов с помощью компьютеров [physical computing], программированию в *Minecraft* и представлению с модулем камеры Pi в главной роли. На второй день курс разбивается на группы, занимающиеся различными проектами, от приложений для Twitter до создания роботов на базе Pibrella и электронных штуковин с Babbage the Bear. Если вам это интересно и вы хотите стать сертифицированным преподавателем по Raspberry Pi, посетите www.raspberrypi.org/picademy и подайте заявку на участие в курсе. LXF

Ключевые этапы в обучении программированию

Ключевой этап 1 (5–7 лет)

Дети учатся понимать алгоритм как серию шагов, требуемых для решения проблемы или достижения цели. Логически мыслить, писать и тестировать простые программы на разных устройствах, таких как планшеты или Raspberry Pi. Задавать программируемым роботам простые инструкции, нажатием клавиш со стрелками или перетаскиванием символов для создания последовательности, которая направляет робота по лабиринту или позволяет обходить препятствия для достижения пункта назначения.

Ключевой этап 2 (7–11 лет)

Дети используют логику и алгоритмическое мышление для решения проблем с использованием фундаментальных понятий программирования. С помощью *Scratch*, бесплатной кросс-платформенной среды разработки, дети учатся создавать алгоритмы, которые усложняются по мере развития их мышления. Создание и программирование модели аллигатора из Lego с инфракрасным датчиком приближения во рту — прекрасный способ узнать о том, что такое выбор, повторение и переменные. Концепт выбора реализуется в программе для контроля за активацией датчика. Добавленная к ней операция тестирования работы датчика позволяет продемонстрировать пользу переменных.

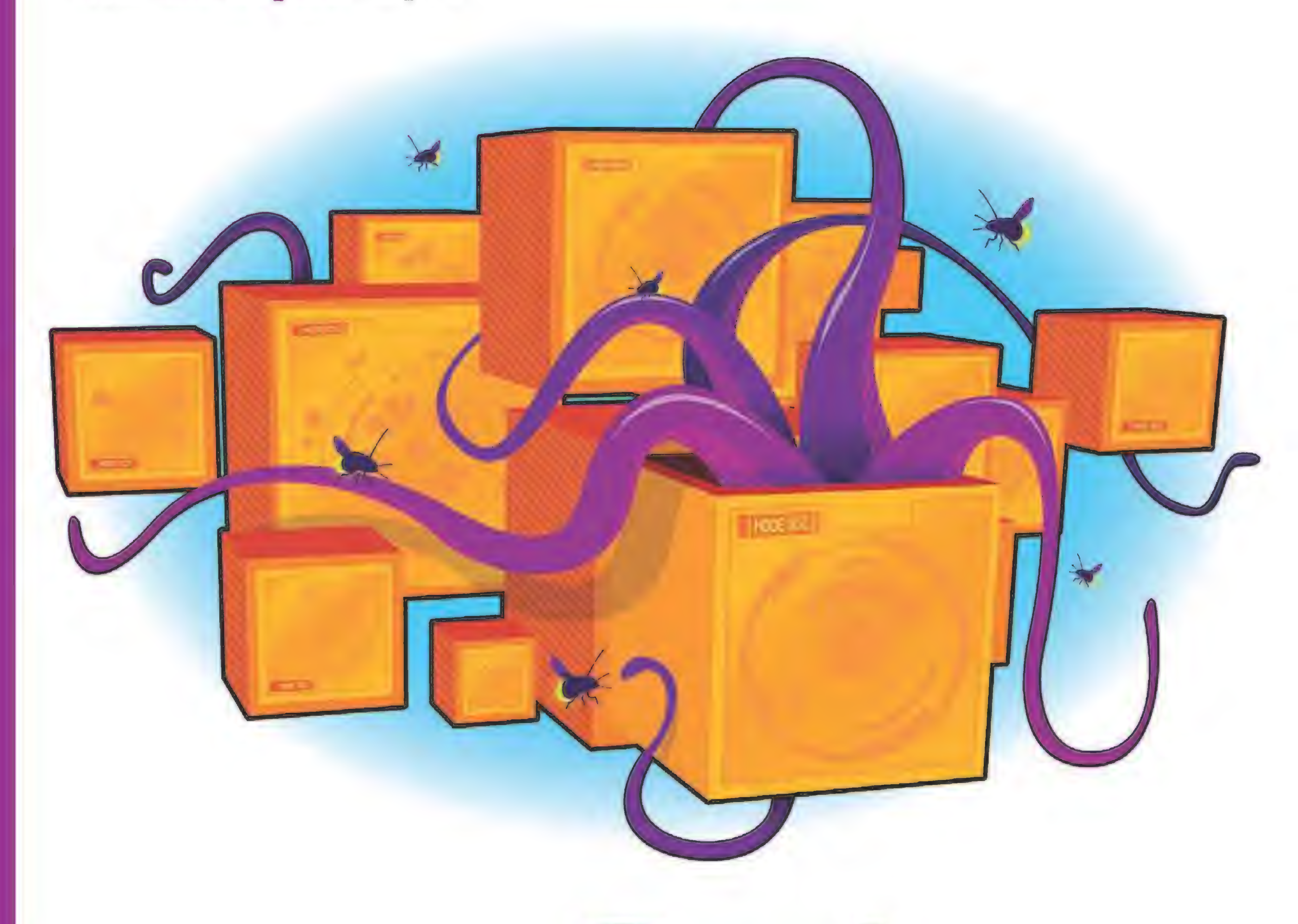
Ключевой этап 3 (11–14 лет)

На этом этапе дети работают с текстовыми языками; зачастую школы выбирают Python. Raspberry Pi B+, где есть порты GPIO для подключения устройств типа Pibrella или Piface — отличное средство, чтобы попытать себя в программировании, сконструировав, например, охранную сигнализацию. Небольшой микроконтроллер Arduino совместим с множеством внешних датчиков, и программировать его можно как с помощью встроенных библиотек Arduino IDE, так и импортированных. Использование различных дистрибутивов Linux дает представление о командной строке, параллельной установке нескольких операционных систем на одной машине и целях обновления, стандартных библиотеках Python и специальных пакетах устройств.

Ключевой этап 4 (14–16)

На этом этапе молодые люди развивают свои вычислительные способности и выбирают пути, которые приведут или не приведут их к сдаче экзаменов для подтверждения своих знаний. Этот период может стать подготовкой к сфере будущего труда или основой для дальнейшего обучения. Популярным проектом может быть дизайн, конструирование и программирование роботов с помощью базового комплекта "Shrimp" для сборки бота с нуля и управления им через Arduino IDE, или один из превосходных комплектов для создания роботов Raspberry Pi.

Те, у кого есть интерес к музыке или искусству, могут проектировать носимые инструменты "Picussive" с тактильными датчиками или выполнить компьютерную инсталляцию, связав датчики и индикаторы с открытыми данными посредством web-интерфейса.



Что за штука... Ceph

Джолион Браун собирает свои объекты, блоки и файлы в открытое хранилище, пришедшее на смену технологии SAN.

В Ceph. Подождите-ка... это не тот парень с большими усами из *Emmerdale*?

О Вы выдаете свой возраст... [Сериал *Emmerdale* впервые вышел в 1972 г.] Нет, *Ceph* — это открытое, унифицированное, распределённое хранилище, демонстрирующее прекрасную производительность, надёжность и масштабируемость. Оно бесплатное и может использоваться на обычных машинах.

В Унифицированное? Как-то это уныло выглядит.

О В унификации ничего дурного нет. Дело в том, что *Ceph* одновременно включает блочную, объектную и системную память. Масштабируется до экзабайтного уровня, не имеет одиночных точек отказа, обладает высокой надёжностью и при этом легко управляется.

В Слишком хорошо, чтобы быть правдой. Как же тогда устроен *Ceph*?

О На верхнем уровне, *Ceph* представляет собой ряд узлов (одиночных машин), объединённых в Ceph Storage Cluster. Как заявлено создателями, кластер *Ceph* может быть «бесконечно» масштабируемым, благодаря тому, что основан на RADOS (Reliable Autonomic Distributed Object Store) и состоит из тысяч узлов. Узлы, в свою очередь, запускают демоны отдельных процессов в программной части *Ceph*. Мониторы *Ceph* поддерживают основную копию карты кластера, *Ceph* OSD Daemons отвечают за операции чтения/записи на дисках хранения и применяют алгоритм под названием CRUSH для определения расположения данных, а серверы метаданных *Ceph* хранят информацию о файловой системе — в частности, имена каталогов и владельцев файлов.

В RADOS, CRUSH — вы, как всегда, сыплете сокращениями, которые для меня все похожи на названия каких-нибудь вражеских организаций, охотящихся на Джеймса Бонда. Что это?

О SPECTRE тут ни при чём, но OK. RADOS — это способ использовать интеллект, заложенный в хранилищах и узлах, а не просто давать им отвечать на IO-запросы, как в более традиционных системах. Все узлы в системе RADOS обладают

доступом к так называемой схеме кластера [Cluster Map], которая отслеживает, какие устройства находятся в кластере, а также задает распределение данных по всем устройствам. Узлы сами контролируют избыточность данных, обнаружение и восстановление после сбоя, что уменьшает число узких мест, возникающих в более централизованной традиционной системе. В кластере может быть много узлов, каждый из которых будет решать собственные задачи, но клиент все равно будет воспринимать кластер как единое целое.

В Я решил погуглить CRUSH, чтобы вас впечатлить. Controlled Replication Under Scalable Hashing, верно?

О Верно! Это алгоритм, который *Ceph* использует для вычисления местоположения объектов, и он во многом обеспечивает масштабируемость Ceph. Проще говоря, при его посредстве клиенты вычисляют, куда должны быть помещены данные, а демоны OSD — где хранить копии этих данных. Информация, организованная таким образом, а не в виде более традиционных списков распределения, позволяет минимизировать избыток метаданных файла. Получение данных и их распределение также сильно упрощаются.

В А схема кластера, о которой вы упоминали, что это?

О Для работы системы, клиенты и демоны *Ceph* должны понимать топологию кластера. Она содержится в схеме кластера — это общее название для пяти отдельных схем. Они хранят информацию о мониторах, OSD, группах размещения [Placement Groups] (на языке *Ceph* именуемых PG и в общих чертах представляющих собой коллекции объектов), серверах метаданных и схему CRUSH. Карта CRUSH — это список устройств хранения данных и правил доступа к ним. Изменение схемы CRUSH — задача администратора *Ceph*. Каждая карта содержит историю внесенных в неё изменений; например, карта обновляется при подключении или отсоединении OSD. Мониторы *Ceph* хранят главную копию схемы кластера.

В Как-то непонятно. Позвольте кое-что для себя прояснить. Допустим, у меня есть кластер *Ceph* и я хочу записать туда файл. Как это сделать?

О Давайте представим, что у вас есть файл, который вы хотите сохранить как объект в своем кластере (на самом деле, *Ceph* внутри всё хранит как объекты). В клиенте *Ceph*, уже некоторое время входящем в состав ядра Linux, есть интерфейс командной строки для хранилища объектов. Кроме того, хранилище объектов *Ceph* может обрабатывать вызовы JSON, что даёт совместимость с Amazon S3 и OpenStack Swift. Клиент принимает имя файла, название, которое вы хотите дать объекту, и пул, где вы хотите его разместить. Пул — это просто логический набор OSD и групп размещения, а также правил их использования. Администратор *Ceph* может, к примеру, выделить каждому пользователю собственный пул. Затем клиент обращается к серверу монитора (их в *Ceph* целый кластер), чтобы получить последнюю копию схемы кластера. Так он получает информацию обо всех мониторах, OSD и серверах метаданных в кластере. С помощью хэшей объекта и ID пулов клиент вычисляет, в какую из групп следует разместить объект и с каким OSD нужно для этого связаться. Затем OSD записывает объект на своем локальном диске в однородное пространство имён — никаких деревьев каталогов здесь нет, так что об этом можно не беспокоиться. Объект получает идентификатор, двоичные данные и метаданные в виде набора пар имя/значение. Содержание метаданных определяется тем, какой используется клиент. OSD затем разместит его копии в местах, которые рассчитает самостоятельно посредством запуска CRUSH, и сообщит клиенту о завершении операции. Самую подробную информацию о том, как это работает, вы можете получить на http://ceph.com/docs/master/architecture.

> «Да, Ceph — это от „цефалопод". Тема моллюсков здесь очень актуальна.»

В Приведите, пожалуйста, пример обработки блочных файлов.

О По сути, внутри всё точно так же. На деле, интерфейс блочного устройства распределяет информацию по всему кластеру, улучшая производительность. У этих устройств экономное («тонкое») отведение памяти, а значит, они будут использовать хранилище только при необходимости, и их можно копировать и клонировать, как в любой современной системе.

В ОК. Я решил построить свой кластер *Ceph* сегодня же вечером. Что мне для этого нужно?

О *Ceph* требует как минимум монитора, а также одну или более запущенных OSD. В примерах, приведённых в практическом руководстве на http://ceph.com/docs/master/start, для запуска используется три или четыре узла. Заметим, что это не обязательно должны быть реальные, физические машины для тестирования: виртуальные хосты тоже подойдут, но может пострадать производительность. При создании более крупных кластеров определённо следует заглянуть в официальную документацию на http://bit.ly/CephDocs. По хорошему, *Ceph* может затребовать довольно мощное — хотя и не самое крутое — оборудование и межсетевые соединения. Всё зависит от цены и сути вопроса. Хотя, чтобы попробовать, его можно легко запустить и с помощью *Vagrant*.

В Пожалуй, я бы не отказался от помощи со всем этим. Куда мне обратиться за дополнительной поддержкой?

О Конечно, всегда есть сообщество *Ceph*. Мало того, Inktank Storage (компания, основанная разработчиком *Ceph*) недавно куплена Red Hat. Они предлагают пакетную версию *Ceph*, известную как *Ceph Enterprise,* и я уверен, что тамошние ребята будут рады поговорить с новым клиентом. Компания открыла доступ ко всем инструментам, которые в Inktank были только платными — отдельно стоит отметить консоль *Ceph* и инструмент управления, *Calamari*.

В *Calamari*? Ага, теперь понятно, почему выбрано имя *Ceph*.

О Да, *Ceph* — это от «цефалопод» [cephalopod — головоногий моллюск, — *прим. пер.*] Тема моллюсков здесь связана с идеей множества компонентов, функционирующих независимо друг от друга — как щупальца у моллюска. *Calamari* — это приложение к кластеру *Ceph*, в виде информационной панели на базе браузера, оно показывает статус различных компонентов в кластере, статистику использования и т. д. В Сети есть множество видео, демонстрирующих, как выглядит *Calamari*; их легко найти, набрав в поиске 'Inktank *Calamari*'. Скачать его можно из репозитория *Ceph* GitHub на https://github.com/ceph. Обратите внимание, что код открыли всего пару месяцев назад, поэтому установка может быть несколько сложнее, чем в других современных программах Open Source. По крайней мере, пока поставщики Linux не доведут до ума пакетную версию.

В Всё это звучит очень многообещающе. Есть ещё какие-то фишки?

О Последняя версия *Ceph* называется Firefly и включает возможность распределения данных в кэше по слоям [cache tiering]. То есть «горячие» данные, в активном состоянии, могут перемещаться на высокоскоростное устройство — типа SSD, а «холодные», не используемые данные отправляются на медленное устройство. *Ceph* также поддерживает кодирование с исправлением ошибок.

В А чего нам следует ждать от *Ceph* в будущем?

О Ну, у Red Hat уже было собственное хранилище, ещё до покупки Inktank, а именно *GlusterFS*. На данный момент не совсем ясно, что будет с обоими проектами — объединят ли их в один или пока оставят как есть. Далее, слабым местом *Ceph* считался элемент для работы с файловой системой. И хотя с каждым релизом он улучшался, вероятно, там ещё есть куда стремиться. Red Hat будет ориентировать *Ceph* на крупных клиентов, так что стоит ожидать появления каких-то «корпоративных функций». Наконец, *Ceph* является большим любимцем пользователей *OpenStack*, так что, скорее всего, он укрепит свои позиции. LXF



По рецептам доктора Брауна

Д-р Крис Браун
Доктор обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.

Эзотерическое системное администрирование из причудливых заворотов кишок серверной

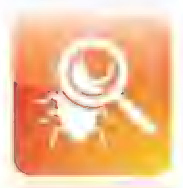












Пронырливый canvas

Анализ ДНК в последние годы произвел революцию в криминалистике, помогая как оправдывать невиновных, так и выносить приговор виновным. Теперь оказывается, что существует и наука (если это слово подходит) о дактилоскопии пользователей Сети. Она включает самые разные действия, от получения вашего IP-адреса и возможного географического положения до извлечения детальной информации о просмотре Сети с помощью JavaScript и анализа стека TCP/IP для определения ОС.

Недавно появились сообщения об использовании новой технологии, получившей название «дактилоскопия с помощью canvas». Она работает примерно так: сайт просит ваш браузер вывести скрытое изображение в элементе HTML5 <canvas>. Обычно изображение содержит несколько перекрывающихся строк одного и того же текста, выполненных разными цветами. Звучит довольно безобидно. Но оказывается, что на прорисовку окончательного изображения на уровне пикселей может повлиять огромное количество факторов: различия в рисунке шрифта, характеристики стека по работе с изображениями, различия в методах сглаживания и прорисовке пикселей и, возможно, многое другое. Передав прорисованное изображение обратно на сервер и вычислив свертку, вы получите подпись, которая с высокой точностью идентифицирует и браузер, и операционную систему, и совершенно не зависит от поля 'HTTP User Agent'.

Если хотите увидеть эту идею в действии, попробуйте зайти на www.browserleaks.com/canvas с невообразимым количеством сочетаний разных браузеров и операционных систем. (На сайте также есть другие примеры использования информации, полученной в результате утечки.) Лично я поместил бы это в категорию «подло, но не страшно». В конце концов, если кому-то интересно, что у меня Chrome 34.0 и Ubuntu 14.04, им достаточно прочесть эту колонку!
chris.linuxformat@gmail.com

SSH издалека

Позарез нужно подключиться к компьютеру из шезлонга на пляже? Не бойтесь, помощь близка.

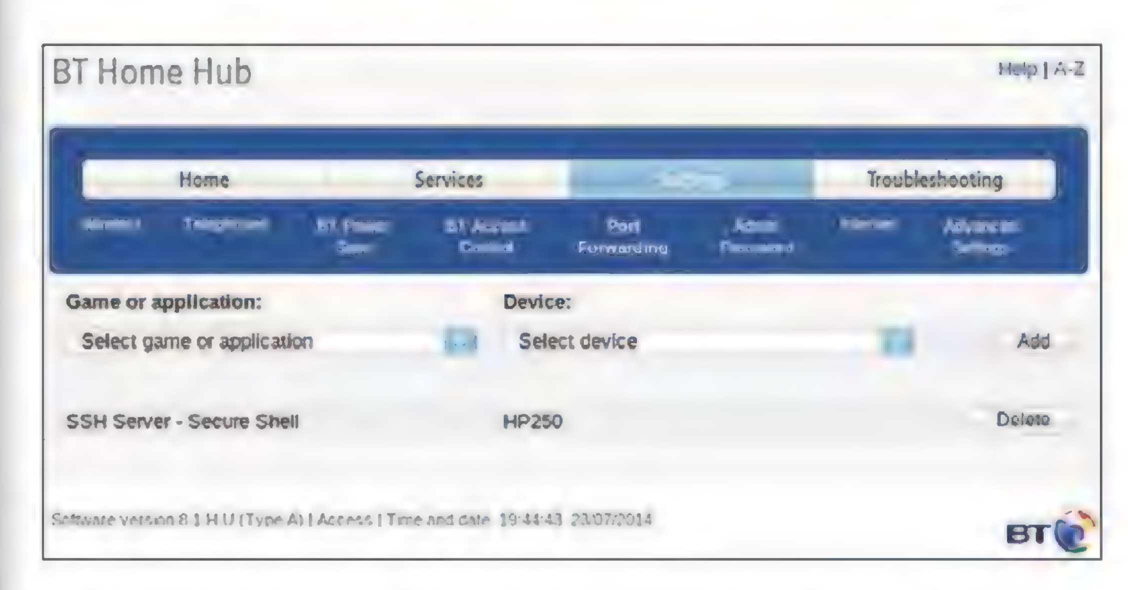
Вы уехали на отдых, и ваша вторая половина запретила брать с собой ноутбук. Все, что у вас есть — телефон с Android, но вам очень нужно подключиться к домашнему компьютеру, и вы в отчаянии... Вот что нужно сделать.

Сначала установите на телефон SSH-клиент из Google Play. Я выбрал *Juice SSH*, за максимальный рейтинг и свежий лимон на иконке; но немало и других. Уверен, что для iPhone тоже есть масса клиентов, хотя точно не знаю — у меня его нет.

Следующий шаг (и его сделать нужно, пока вы не уехали) — включить перенаправление портов на роутере, чтобы порт 22 внешнего сетевого интерфейса роутера перенаправлялся на порт 22 компьютера, к которому вы хотите подключиться. Если у вашего компьютера есть собственный внешний IP-адрес (что вряд ли), можете пропустить следующий шаг, но если компьютер находится во внутренней сети с частным IP-адресом вроде 192.168.что-нибудь, то его придется сделать. Точные действия зависят от производителя и модели используемого роутера. У моего для настройки используется встроенный web-сервер, слушающий адрес 192.168.1.254. Зайдите на него, переверните роутер, отклейте бумажку с паролем администратора и войдите в систему под этим паролем. Откройте соответствующую панель с настройками и включите перенаправление порта SSH на свой компьютер (см. стр. 28).

Затем убедитесь, что знаете свой внешний IP-адрес. Его можно найти в настройках роутера, но быстрее будет зайти на сайт вроде http://whatismyipaddress.com, который сообщит вам не только IP-адрес, но и провайдера, географическое положение и цвет глаз. (Ну, последнее-то я придумал.) Этот адрес потребуется для настройки подключения на телефоне.

Теперь все подготовлено. Запустив передачу данных на телефоне, вы должны успешно подключиться. Главное, не проговоритесь случайно об этом своей второй половине!



› На домашнем роутере нужно включить перенаправление портов.

Интеграция Linux и Windows

Как на практике выглядит миграция большой компании из Windows в Linux и какие инструменты упростят переход?

Несколько лет назад я читал учебные курсы по Linux компании Bloomberg (предоставляющей финансовые информационные услуги) в Лондоне. У них было полдюжины старых серверов, которые больше не отвечали минимальным требованиям последней версии Windows Server, и они решили выбросить их в мусорку. Потом кто-то предложил поставить на них Linux и запустить *Samba*, которая предоставляла бы доступ к файлам на серверах с компьютеров с Windows. Что они и сделали. Это прекрасный способ внедрения Linux в корпоративное окружение. Так как *Samba* говорит на языке родных протоколов Windows (SMB/CIFS), то пользователи настольных компьютеров Windows едва ли заметили разницу, и переучивать их не пришлось.

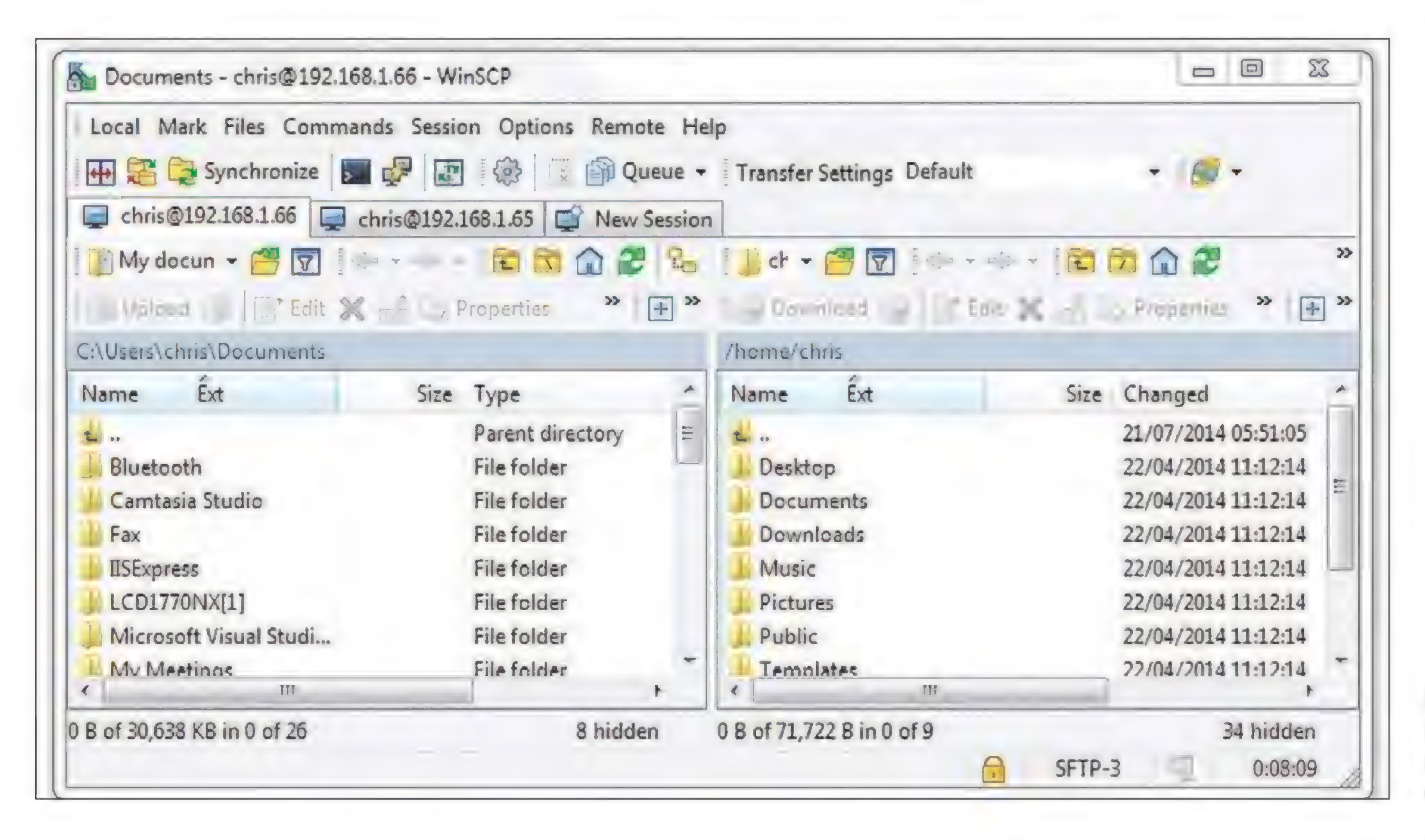
Перевести настольные компьютеры на Linux — дело другое. В 2004 году, после того как Novell купила SUSE, я получил контракт на проведение однодневного учебного занятия в европейских офисах Novell. В общей сложности я повторил это занятие около 20 раз для аудитории, состоявшей в основном из людей нетехнического склада — администраторов, офисных менеджеров, менеджеров по продажам и т. д. Разнообразие откликов, которые я увидел, было интересным. Некоторые, узнав, что их компания переходит на Linux были готовы принять новую технологию с распростертыми объятиями. Другие высокомерно не хотели тратить день на изучение чего-то нового, хотя, подозреваю, за их высокомерием на самом деле скрывалась боязнь не справиться с изменениями. И именно об этом беспокоилось большинство людей: о выходе из зоны комфорта и о том, смогут ли они продолжить делать то, что в совершенстве делали в знакомых системах Windows.

У одной милой немки, с которой я разговаривал (она работала персональным ассистентом руководителя и скоро уходила на пенсию), была более взвешенная историческая перспектива. «Я начинала работать на печатной машинке, потом появились эти машинки с шаровой головкой IBM Selectric, потом компьютеры с DOS и *Wordstar*, — говорила она. — Я пользовалась примерно пятью версиями Windows и *Microsoft Office*, а теперь это Linux и *OpenOffice*... Когда я смогу закончить изучать новое и просто работать?»

Некоторые сотрудники имели более конкретные опасения. Особенно беспокоились сотрудники финансового отдела, из-за сильной привычки к приложениям, не имевшим эквивалентов в Linux. Другие, успешно перейдя с *Microsoft Office* на *OpenOffice*, беспокоились о необходимости обмена офисными документами с клиентами, которые по большей части все еще были в мире Windows. Novell устроила некую сортировку приложений: сотрудники могли проголосовать за приложения, которые были для них наиболее критичны. Подсчитав голоса, компания решала, направить ли свои усилия на замену или на портирование этих приложений. В этих обстоятельствах частью решения могли стать слои совместимости, такие как *Wine*, позволяющие запускать приложения Windows в Linux. »

Интероперабельность X Window

Как вы, возможно, знаете, X Window System по сути представляет собой клиент–серверную архитектуру, которая позволяет графическому приложений подключаться к удаленному X-серверу для управления его интерфейсом пользователя. Для Windows доступны несколько X-серверов, и это еще один способ запуска в Linux графических приложений рабочего стола Windows. Популярные варианты — *Xming* и *MobaXterm*; также есть X-сервер в составе *Cygwin*. При подключении по недоверенной сети обязательно подумайте о безопасности, так как ваш ввод с клавиатуры и содержимое окна могут быть перехвачены. В SSH эта проблема обычно решается пропусканием трафика к X-серверу через зашифрованное соединение.



› Обменивайтесь файлами защищенно с помощью *winscp*.

Еще до перехода на Linux Novell перевела пользователей на *OpenOffice* (я не участвовал в этом, но могу представить, что это вызвало большее сопротивление пользователей по сравнению с изменением операционной системы). И это интересный момент: для перехода к открытым программам не обязательно отказываться от Microsoft. Существует масса крупных открытых приложений, которые работают в Windows — *LibreOffice*, *Apache*, *MySQL*, *GIMP*, *Scribus*, *Chrome*, *Audacity*... список длинный.

Как я раньше отмечал в этом журнале, точную информацию о степени принятия Linux получить трудно. (Я имею в виду широкое принятие в серверном и настольном окружении, а не смартфоны и не полностью занятый Linux мир суперкомпьютеров.) Наверное, в Red Hat знают, сколько подписок они продали, но для большинства дистрибутивов одна загрузка ISO-образа может привести к нескольким сотням установок, а может не привести ни к одной. Есть много опросов и «докладов», но большинство из них проводятся в угоду какой-то точке зрения и не дают сбалансированной картины. Очевидно, что большинство компаний в значительной мере приняли Linux (не считая тех компаний, которые перешли на него полностью), и это означает, что в таких организациях Linux и Windows в обозримом будущем должны сосуществовать. И поэтому последние три месяца я пишу в своей рубрике об интеграции Linux и Windows. Я сосредоточился на их взаимодействии, используя *Samba* для доступа к файлам и для интеграции учетных записей в Active Directory Windows.

Средства удаленного доступа

Поэтому я хочу немного сместить акцент и представить несколько утилит для доступа к удаленному компьютеру Linux с локального рабочего стола Windows. Мы начнем с утилит доступа к командной строке, затем перейдем к утилитам доступа к рабочему столу.

Очевидный протокол для доступа к командной строке — SSH. (Если кто-то еще пользуется Telnet, пожалуйста, идите и сдайте его в местный полицейский участок — я устрою вам амнистию.) Использование SSH для связи Linux–Linux обширно освещалось в этом журнале [см. «Учебники» **LXF178/179**, стр. 78], и я не буду обсуждать это снова. Но как насчет Windows? Существует несколько SSH-клиентов для Windows, и я расскажу о трех. Во-первых, *Bitvise Tunnelier* (www.bitvise.com/ssh-client). Наряду с клиентом SSH *Tunnelier* содержит эмулятор *xterm* или *vt100* и графический клиент *SFTP*. Последний представляет собой обычный двухпанельный менеджер с локальными файлами на левой панели и удаленными на правой. Затем есть *Cygwin*, к которому я вернусь через минуту; но самым популярным клиентом SSH для Windows, кажется, является *PuTTY* (для его загрузки нажмите на ссылку PuTTY на www.chiark.greenend.org.uk).



› **Самая популярная утилита просмотра VNC, *RealVNC*, позволяет подключиться к своему любимому рабочему столу Linux из Windows.**

У *PuTTY* есть несколько окон с настройками, на полное исследование которых потребуется время (мне одному кажется, что утилите для доступа с командной строки странно иметь графические окна с настройками?), но единственное, что нужно обязательно указать — IP-адрес компьютера (с Linux), к которому вы подключаетесь. У вас будут запрошены имя пользователя и пароль для входа в систему, после ввода которых откроется командная строка Linux-компьютера.

Cygwin — программа другого рода. Она приносит в Windows значительное количество открытых утилит. Клиент SSH — лишь одна из многих. Небольшая выборка: категория пакетов 'Base [Базовые]' включает *Bash*, все основные утилиты GNU, *file*, *gawk*, *grep*, *gzip*, *man*, *sed*, *tar* и довольно много других. Категория 'Editors [Редакторы]' включает emacs, gvim, joe, nano, ted и vi. Категория 'interpreters [Интерпретаторы]' включает *Emacs*, *Gawk*, *Python*, *Ruby*, *Perl*, *Tcl*. Категория 'Net [Сеть]' содержит *bind*, *curl*, *openssh* (включая важный для нас клиент SSH), а также web-серверы *Apache* и *Lighttpd* плюс *Squid*. Список опять-таки длинный, но, надеюсь, идею вы поняли: здесь много хорошего! Программа установки позволяет выбрать и установить нужные пакеты, а если захотите добавить, сможете запустить ее позднее. (Загрузите пакет с www.cygwin.com.)

Pageant

В утилитах вроде *WinSCP* и *PuTTY*, выполняющих аутентификацию по SSH, можно настроить аутентификацию с помощью закрытого/открытого ключей как альтернативу парольной аутентификации. Для генерации ключевой пары можно воспользоваться *PuTTYgen* (по аналогии с *ssh-keygen* в Linux), а в качестве агента аутентификации взять *Pageant*. Он хранит закрытые ключи в памяти и предоставляет их по требованию, и вы сможете аутентифицироваться без пароля и парольной фразы, необходимой для разблокирования закрытого ключа. Эта программа аналогична *ssh-agent* в Linux. Подробности см. на www.winscp.net/eng/docs/ui_pageant.

Cygwin и *PuTTY* предоставляют совершенно разные возможности. Для меня *PuTTY* — лишь способ как можно быстрее избавиться от Windows и подключиться к Linux. *Cygwin*, с другой стороны, создает для меня гораздо более полное окружение в стиле Linux. В нем есть большинство моих любимых утилит, и я могу счастливо сидеть в командной строке *Cygwin*, воображая, что это на самом деле Linux. Я подробно рассказывал о *Cygwin* в **LXF180** (см. стр. 68).

Удаленный доступ к рабочему столу

Но, возможно, вас заботит, как подключиться к рабочему столу Linux с компьютера Windows? В мире есть огромное количество утилит для этого. (Для сравнения см. http://bit.ly/**LXF**wiki.) Одна из самых известных — VNC, сокращение от довольно невразумительного "Virtual Network Computing [Виртуальные сетевые вычисления]". Иногда мне хочется, чтобы люди, которые придумывают эти названия, уделили бы минутку выбору такого, которое подсказывало бы, что приложение на самом деле делает... но я отвлекся. В этой программе используется протокол RFB (Remote Frame Buffer — удаленный буфер кадра).

Общая идея VNC в том, что на удаленном компьютере, к которому вы хотите подключиться, запускается сервер VNC, а затем на локальном компьютере запускается программа для просмотра (клиент) VNC. Протокол VNC передает изменения экрана с удаленного компьютера на локальный, и в зависимости от его настроек на удаленный компьютер также будут передаваться клавиатурный ввод и мышь, как если бы вы сидели за удаленным компьютером. Я говорю «в зависимости от», потому что некоторые серверы VNC поддерживают только режим «просмотра»: в этом режиме можно только наблюдать за тем, что происходит на удаленном компьютере, не взаимодействуя с ним. Это бывает удобно, когда коллеге, работающему на расстоянии, надо что-то продемонстрировать вам.

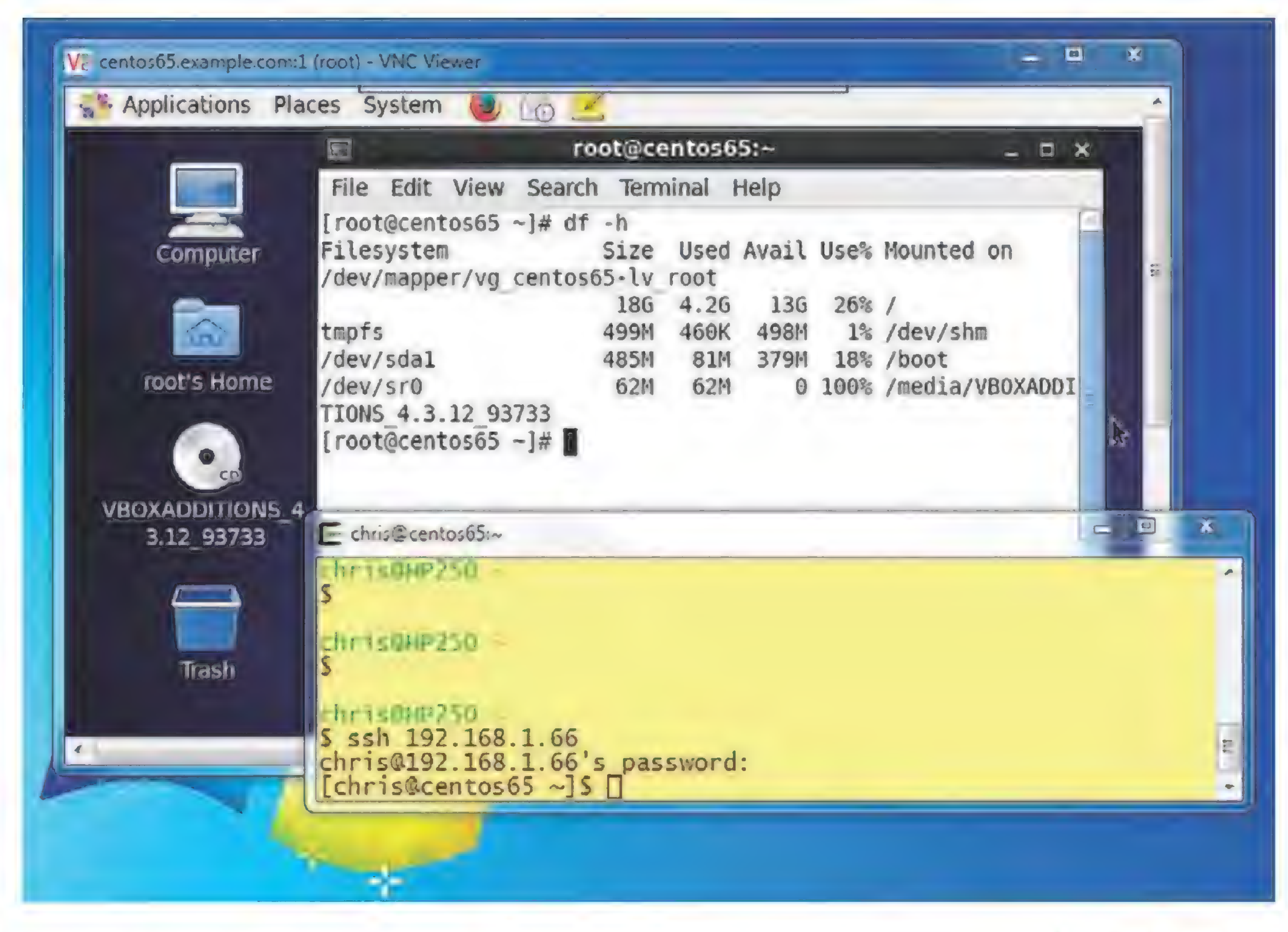
Для иллюстрации идеи я начал с установки просмотрщика *RealVNC* на Windows 7 (www.realvnc.com/download). Первым VNC-сервером, который я попробовал установить, был *Vino*. В Ubuntu 14.04 он является частью рабочего стола Unity/Gnome. Для его запуска поищите "vino" или "desktop sharing" в Dash. У сервера очень простое окно настройки (см. рис. на стр. 67), но у меня

возникли проблемы при попытке подключиться к нему из *RealVNC*: появилось сообщение об ошибке "Unable to connect to VNC Server using your chosen security setting [Невозможно подключиться к серверу VNC с заданными настройками безопасности]". Чтобы это обойти, можно выключить шифрование на сервере заклинанием

```
gsettings set org.gnome.Vino require-encryption false
```

Чтобы сделать эти изменения постоянными, просто откройте файл **/usr/share/glib-2.0/schemas/org.gnome.Vino.gschema.xml**, найдите параметр "require-encryption" и установите его в false. Вообще-то я не люблю отключать функции, связанные с безопасностью, и исследовал бы проблему более тщательно, столкнись я с этим в реальном мире.

Насколько я знаю, в *Vino* можно обращаться к общим ресурсам только для текущего рабочего стола; он не позволяет запускать новые сеансы рабочего стола.



› Рабочий стол Windows с VNC-подключением к компьютеру с CentOS и SSH-подключением к тому же компьютеру в *Cygwin*.

Существует масса других реализаций как сервера, так и клиента VNC — и для Windows, и для Linux. Количество возможных сочетаний клиента и сервера может стать огромным, но самый известный из них — пожалуй, *RealVNC*.

Я также попробовал установить пакеты *tigervnc* (утилиту просмотра) и *tigervnc-server* (сервер) в стандартную версию CentOS 6.5. Эта реализация довольно сильно отличается от *Vino*. Главная программа здесь — *Xvnc*, по сути это X-сервер. Но вместо рисования в локальный буфер кадра, как обычно делает X-сервер, *Xvnc* посредством протокола RFB обновляет содержимое окна программы просмотра VNC. Перед запуском сервера нужно задать пароль командой **vncpasswd**. У каждого пользователя может быть собственный пароль; пароли хранятся в **~/.vnc/passwd**.

Сервер обычно запускается скриптом-оберткой под названием *vncserver*. Он должен прекрасно работать, если запустить его без аргументов:

```
vncserver
```

При каждом выполнении команды **vncserver** запускается новый экземпляр *Xvnc*. Первый слушает порт 5901, второй — 5902, и т.д. (На языке X windows это дисплей :1.0 и :2.0 соответственно.) На каждом сервере также создается новый сеанс рабочего стола. Так несколько пользователей могут подключаться к различным сеансам одного и того же рабочего стола на одном компьютере. Я также протестировал это в Windows с *RealVNC* Viewer. Просто запустите его, и он запросит у вас IP-адрес и номер порта сервера, а затем пароль.

В *Vino* и *TigerVNC* используются различные схемы доступа. В *Vino* рабочий стол разделяется между всеми пользователями по сценарию «дай я тебе покажу». В *TigerVNC* создаются новые рабочие столы, что делает возможной одновременную работу множества пользователей, количество которых ограничивается доступной памятью и производительностью процессора сервера.

Если у вас есть старые номера, я рассказывал о VNC чуть подробнее в **LXF161** [«Рецепты д-ра Брауна», стр. 59].

У меня стремительно заканчивается место, но его еще хватит на упоминание *WinSCP*, свободного (лицензия GPL) графического FTP- и SFTP-клиента для Windows. Чтобы его загрузить, зайдите на www.winscp.net/eng/download.php и щелкните по ссылке 'Installation package [Установочный пакет]'. Интерфейс *WinSCP* имеет как режим проводника (с одним окном), так и режим файлового менеджера — двухпанельный режим, при котором на левой панели отображаются файлы на локальном компьютере (с Windows), а на правой — на удаленном компьютере (с Linux), и файлы просто перетаскиваются между панелями. В этом режиме есть вкладки, на каждой вкладке можно открыть отдельное удаленное подключение.

Можно даже редактировать файлы на удаленном компьютере. Для этого щелкните на них правой кнопкой мыши и выберите Edit [Редактировать]. При этом файл скопируется на компьютер с Windows и откроется во внутреннем редакторе. При сохранении файла он снова скопируется на удаленный компьютер. Сам редактор довольно примитивный, но это явно лучше, чем копировать файлы Linux на флэшку для переноса в Windows, чтобы отредактировать их в *Notepad*. К сожалению, мне на учебных курсах еще попадаются студенты, которые предпочитают так и делать — вместо того, чтобы выучить несколько команд в родном редакторе Linux. Прошу извинить, если кто-то после прочтения этих строк ударился в слезы. LXF

Wine

Для ребят из финансового отдела Novell, привязанных к приложениям Windows, не имеющим открытого эквивалента, решением могла стать *Wine*. Нет, я не имею в виду, что они забудут о своих неприятностях, пропустив стаканчик «Мерло». *Wine* — библиотека совместимости, которая эмулирует вызовы Windows API с помощью эквивалентов Linux, позволяя приложениям Windows запускаться в Linux — до некоторой степени. Эмуляция несовершенна. Джонни Бидвелл затрагивал Wine в **LXF187** [См. «Как Windows живет в Linux, стр. 46], хотя он делал акцент на играх (а никто не играет в игры на работе, правда?). Хотя *Wine* бесплатна, можно облегчить себе жизнь, купив *CrossOver Linux* (£38 от CodeWeavers).

LINUX FORMAT Учебники

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Подтвердивший свою квалификацию физик.

За малым размером стоит само будущее

Фаррел: У нас есть что-нибудь вроде плана?
МакКлейн: Найти Люси и убить остальных.
«Крепкий орешек 4»

Все знают, что динозавры вымерли. Есть масса спекуляций на тему, почему это произошло. Но вымирание было отнюдь не внезапным и началось вообще-то за миллион лет до времени Ч, когда динозавры исчезли. Де-факто темп вымирания не особо менялся, просто перестали образовываться новые виды. Дело, возможно, в том, что малый размер, на который динозавры не шибко-то и претендовали, заняли млекопитающие. Да, мелкота — это «не круто», но именно там обкатываются новые идеи, способные дать интересный результат. Без малых размеров развитие, конечно, не останавливается, но полезные идеи появляются гораздо реже.

Как это соотнести с Linux? Сверхбольшой размер — уже за Linux, что видно по списку Топ-500 суперкомпьютеров; но выросло это из малых размеров благодаря небольшому количеству удачных решений и лицензии. Чтобы захватить средний размер, вытеснив заживившихся динозавров, нужны другие удачные идеи, а их проще обкатать в мелком размере, который тоже явно за Linux. Мне не очень нравится путь Android; для экспериментов лучше выбрать Raspberry Pi — по цене игрушки, хоть и для относительно взрослых и свободных индивидов. Так победим!

E.M.Baldin@inp.nsk.su

В этом месяце вы научитесь...

















И новичкам, и гуру! Всегда полезно будет познать нечто доселе неведомое

Эмуляция: Шанс для ретро-игр

Маянк Шарма надевает розовые очки ради пересмотра классических платформ вчерашнего дня — и благодарит судьбу за то, что эти платформы сошли на нет!



Наш эксперт

Маянк Шарма — бывший пишущий редактор Linux.com, он любит побаловаться с Raspberry Pi.



Неважно, хотите ли вы оживить прошлое или исследовать происхождение игр, недоигранных вами в юности — ваш современный ПК запросто вернет вас к заре компьютерных технологий. В наши дни у средней настольной системы достаточно прыти, чтобы имитировать любой классический компьютер. Итак, мы поможем вам воссоздать классические компьютерные платформы на вашем ПК с помощью эмуляторов с открытым кодом.

Говоря попросту, эмулятор — это программа, создающая виртуальное оборудование, на котором можно загрузить и запустить старинное ПО. Эмулятор создает виртуальные версии сравнительно простого оборудования классических компьютеров. Затем вы можете использовать это виртуальное оборудование, чтобы загрузить старинные операционные системы и программы. С точки зрения программ, они работают на старинной же машине.

Начнем с BBC Micro. Это была серия компьютеров, разработанная компанией Acorn Computer для проекта BBC Computer Literacy Project. Он был запущен в 1981 году и обрел невероятную популярность в Великобритании. Итак, начнем с того, что найдем нужные зависимости:

```
sudo apt-get install libsdl1.2-dev libgtk2.0-dev build-essential
```

Далее, вам нужно будет создать директорию под названием **beebem** и скачать исходные файлы и заплатки в эту директорию:

```
wget http://beebem-unix.bbcmicro.com/download/beebem-0.0.13{.tar.gz,_64bit.patch,-keys.patch,_menu_crash.patch}
```

Распакуйте исходный код командой

```
tar -xzf beebem-0.0.13.tar.gz
```

и примените заплатки:

```
for file in *.patch; do patch -p0 < $file; done
```

Перейдите в распакованную директорию исходного кода с помощью

```
cd beebem-0.0.13
```

и скомпилируйте эмулятор:

```
./configure --enable-econet
```

Эмулируйте старые компьютеры на Raspberry Pi

Будучи одноплатным компьютером, Raspberry Pi очень похож на классические самопальные компьютеры DIY 1970-х и 1980-х, с той лишь разницей, что он куда мощнее, более компактный и имеет лучшие возможности для программирования. И вы можете использовать Pi, чтобы эмулировать первые компьютеры того времени, когда вся эта область была во младенчестве.

Большинство эмуляторов, о которых мы расскажем на нашем уроке, также доступны в репозиториях дистрибутива Raspbian для Pi. Вы можете следовать уроку и установить их сами. Есть также дистрибутив ChameleonPi — это ремикс Raspbian, созданный для эмуляции старых компьютеров, аркад и консолей.

Скачайте дистрибутив с сайта Chameleon (http://chameleon.enging.com) и распакуйте с помощью **bunzip2chameleon.v032.img.bz2**. Теперь запишите образ на SD-карту с помощью инструмента *Win32DiskImager*, вставьте карту в Pi и включите Pi. Войдите в дистрибутив, используя в качестве имени пользователя zx, а в качестве пароля — spectrum. Чтобы скопировать ROM на карту SD, подключите карту к компьютеру и скопируйте их в раздел с меткой 'roms'.



› **Дистрибутив ChameleonPi не предлагает графическую среду, но использует индивидуальную программу запуска с клавиатуры для эмуляторов.**

Эмулируйте классические проприетарные ОС

Есть несколько эмуляторов, способных воссоздать проприетарные платформы конца 1990-х. Если вы хотите поиграть в старые DOS'овские игры на современном компьютере, найдите эмулятор *DOSBox* — он имитирует среду DOS, которая работала на старом оборудовании. Приложение есть в репозитории Universe в Ubuntu. Включив репозиторий, вы сможете установить эмулятор:

```
sudo apt-get install dosbox
```

Запустите эмулятор, введя в терминале **dosbox**. Перед вами откроется нечто вроде еще одного окна терминала с приглашением Z:\. Это *DOSBox*. Вы можете смонтировать локальную папку на своем Ubuntu с помощью команды **mount** (например, **mount C: /home/bodhi**). Она смонтирует вашу папку **home** на диск C:. Теперь скачайте любую игру или утилиту DOS из Интернета в свою директорию **home** и выполняйте их через DOSBox, как если бы вы были на машине с DOS.

Есть еще эмулятор *Basilisk II* для воссоздания Mac Classic и Mac II. Вы можете установить его из репозиториев Ubuntu командой

```
sudo apt-get install basilisk2
```

После установки запускайте его, введя в терминале **BasiliskII**. Но чтобы использовать этот эмулятор, нужен действующий образ Macintosh ROM и диски программы установки MacOS. Вы можете скачать их из официальных репозиториев старого ПО Apple: http://bit.ly/1atDE2x.



› Оживите эпоху DOS, играя в культовую классику вроде *Blood* на *DOSBox*.

затем — **make** и, наконец, **sudo make install-strip**. Когда эмулятор установится, можете вызывать его по **beebem welcome.ssd**.

Далее идет ZX Spectrum, известный также как Speccy. Он был выпущен в 1982 году Sinclair Research Ltd и был среди первых популярных домашних компьютеров в Великобритании. У него имелось восемь разных моделей, и по всему миру было продано более пяти миллионов штук. Есть масса эмуляторов ZX Spectrum, в том числе *FUSE* (Free Unix System Emulator), который вы найдете в репозиториях Ubuntu. Он устанавливается с помощью

```
sudo apt-get install fuse-emulator-common
```

Затем можно взять ROM и утилиты:

```
sudo apt-get install spectrum-roms fuse-emulator-utils
```

Следуйте нашим инструкциям (внизу).

Пока ZX Spectrum обретал популярность в Великобритании, за океаном набирал обороты Commodore 64. Commodore 64 нашел приют в 15 миллионов домов и попал в Книгу рекордов Гиннеса как самая продаваемая модель компьютеров всех времен.

Одним из самых популярных эмуляторов для Commodore 64 является *VICE*, *Versatile Commodore Emulator*, который сыплет релизами с 1993 года. Помимо Commodore 64, *VICE* способен эмулировать и некоторые другие компьютеры Commodore, такие, как VIC-20 и PET. Он тоже имеется в репозиториях Ubuntu, и вы можете его установить с помощью

```
sudo apt-get install vice
```

Однако в версию в репозиториях Ubuntu не включено ядро, поскольку оно содержит код, защищенный копирайтом. Вам придется вручную скачать ядро (и пару других файлов) из исходного кода эмулятора *VICE*. Для этого возьмите исходный код эмулятора:

```
wget http://zimmers.net/anonftp/pub/cbm/crossplatform/emulators/VICE/vice-2.4.tar.gz
```

Затем распакуйте его:

```
tar -zxvf vice-2.4.tar.gz
```

Перейдите в распакованную директорию и измените право собственности на весь файл исходника:

```
sudo chown <username> *
```

Замените <username> на свое имя пользователя. Теперь перейдите в директорию **data/C64** и измените разрешения для файлов: »

Играйте в игры ZX Spectrum

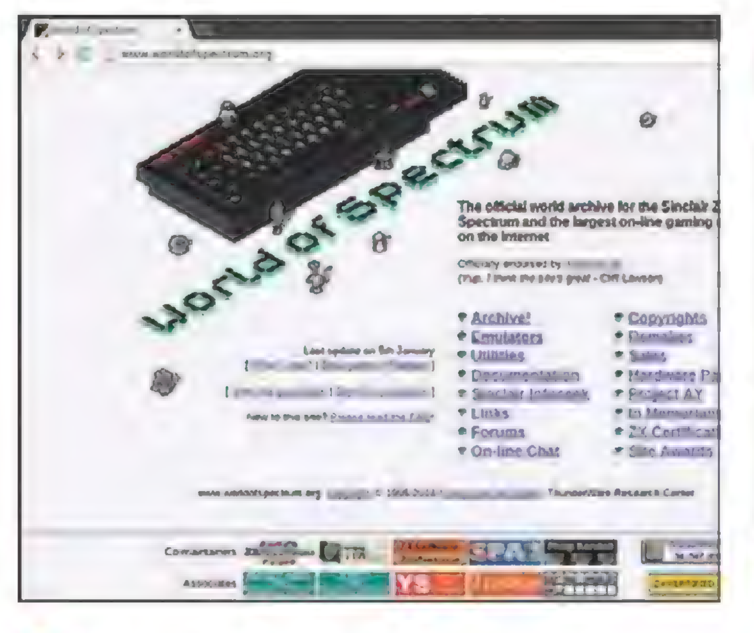


1 Запустите его

Если вы следовали уроку, вы уже установили графический интерфейс для программы установки *FUSE*. Выведите его, введя **fuse** в Unity Dash или в терминале. Если вы на Raspberry Pi, выведите звук через порт HDMI:

```
sudo amixer cset numid=3 2
```

Чтобы вывести звук через аудиопорт, следует заменить 2 на 1.



2 Скачайте игры

Вы можете законнейшим образом скачать игры для Spectrum из множества источников, таких как www.worldofspectrum.org. Все игры на этом сайте доступны в виде сжатых архивов, которые при распаковывании дадут вам файл **.tap**.

Перед загрузкой игры перейдите в Help > Keyboard, чтобы познакомиться с раскладкой клавиатуры Spectrum.



3 Загружайте и играйте

Чтобы начать игру, из меню перейдите в Media > Tape > Open, а затем перейдите в распакованный **.tap**-файл игры. Далее, в окне Spectrum нажмите на клавишу J на клавиатуре, чтобы загрузить игру, потом Ctrl+P, а после этого нажмите Enter — и можете приступать к игре.

Когда вам надоест играть, перейдите в Machine > Reset, чтобы перезагрузить эмулятор.

```
sudo chmod a+rw basic kernal chargen
```

Затем скопируйте их в исходную директорию *VICE*:

```
sudo cp basic chargen kernal /usr/lib/vice/C64
```

Теперь вы можете запустить эмулятор с помощью команды **x64**.

В середине 1980-х Atari выпустили Atari ST. ST означает sixteen/thirty-two [шестнадцать/тридцать два]; это ссылка на его микропроцессор Motorola 16/32-bit. Atari ST был одним из первых домашних компьютеров, у которых имелось 512 КБ ОЗУ, GUI и флоппи-дисковод.

Отличный эмулятор ST — *Hatari*, он имеется в репозиториях пакетов Ubuntu и устанавливается так:

```
sudo apt-get install hatari
```

Перед тем, как вы его загрузите, *Hatari* потребует у вас файл образа TOS. TOS — это операционная система, встроенная в ST. Поскольку она защищена копирайтом, вам придется создавать файл образа из исходного Atari ST ROM с помощью таких программ, как *TOSDUMP.PRG*.

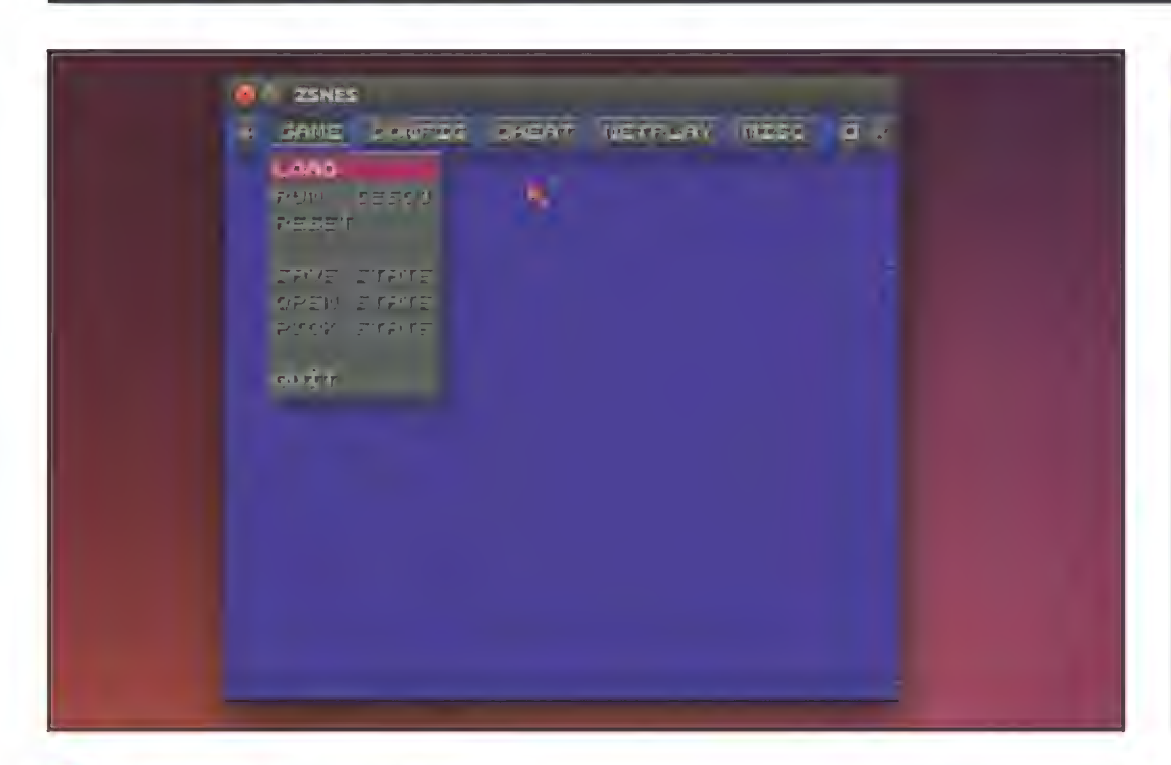
Если вам не добыть настоящий образ TOS, можете попробовать образ EmuTOS (http://emutos.sourceforge.net/en). Скачайте и скопируйте его в **/usr/share/hatari/tos.img** (по умолчанию *Hatari* именно там рассчитывает найти файл образа). Когда вы запустите *Hatari*, выведется классический зеленый рабочий стол Atari ST, *GEM*.

И, наконец, Commodore Amiga, появившийся через пару месяцев после Atari ST. Amiga был очень популярен, и многие считают его первым домашним компьютером с возможностью работы с мультимедиа и многозадачностью [так и было, — *ред.*]. Найдите эмулятор для Amiga в репозиториях Ubuntu:

```
sudo apt-get install fs-uae-launcher
```

Установив эмулятор, следуйте инструкциям, чтобы система Amiga установилась и заработала. LXF

Эмулируйте SNES



1 Скачайте эмулятор

Super Nintendo Entertainment System — бесспорно, самая потрясающая консоль эпохи, предшествующей появлению 3D, и у нее есть очень зрелый эмулятор под названием *ZSNES*, который вы можете установить из репозиториев:

```
sudo apt-get install zsnes
```

Можете запустить его, введя **zsnes** в Unity Dash или терминале.



2 Скачайте игру

Распространение образов ROM является делом темным с юридической точки зрения, именно поэтому проект ZSNES не предоставляет вам образов ROM. Однако вы можете найти сайты с образами ROM для Super Nintendo. Один из них — www.virtualsupernes.com, и он утверждает, что предоставляет образы ROM вполне легально.



3 Настройте клавиатуру

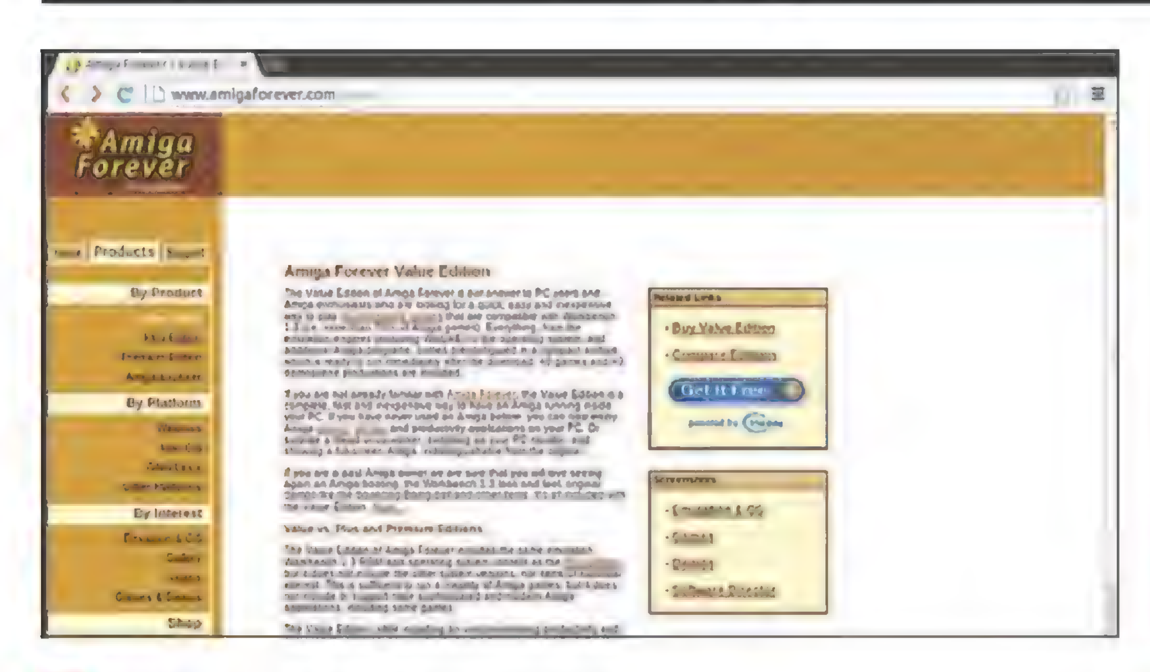
Прежде чем начать играть, неплохо было бы настроить клавиатуру. Перейдите в Config > Input. В открывшемся окне выберите опцию Keyboard/Gamepad и нажмите на кнопку Set Keys. Появится крошечное окно, предлагающее вам нажать кнопку для соответствующего действия в SNES.



4 Загрузите и играйте

Найдя файл ROM для игры, перейдите в Game > Load, выберите файл SMC для игры и нажмите на кнопку Load. Во время игры нажимайте на клавишу Esc, чтобы сделать паузу и вывести главный экран *ZSNES* с меню. Для сохранения текущего статуса перейдите в Game > Save State.

Устанавливаем Amiga Workbench OS



1 Собираем файлы

Эмулятор Amiga требует как прошивки Kickstart, так и файлов Workbench. Есть несколько возможностей на законном основании получить эти файлы; вы можете или купить пакет *Amiga Forever* от Cloanto (www.amigaforever.com), поставляемый с лицензированными ROM, или извлечь Kickstart ROM с вашего настоящего Amiga с помощью инструмента типа *TransROM* или *GrabKick*.



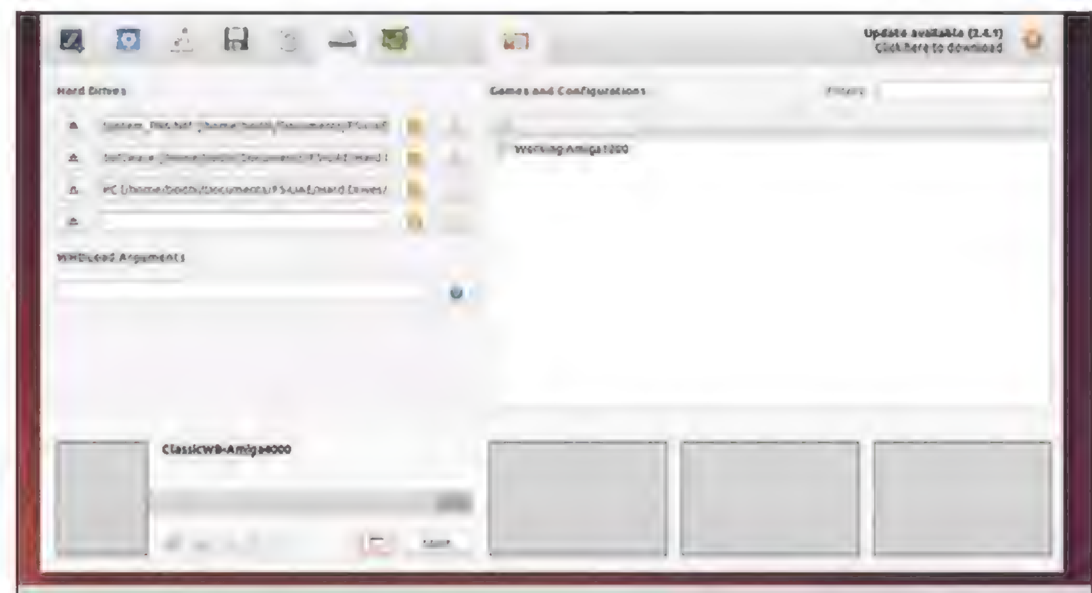
2 Скачайте ClassicWB

Мы собираемся установить свободно доступную операционную систему ClassicWB Workbench на наш эмулятор Amiga. Есть разные виды Classic Amiga Wokbench, включая разновидность, специально созданную для эмуляторов UAE. Вы можете скачать сжатую ОС размером 177 МБ с сайта http://classicwb.abime.net/classicweb/downloaduae.htm.



3 Перенесите файлы

Далее, нужно создать пустую папку в **Documents/FS-UAE/Hard Drives/** под названием **ClassicWB**, а внутри этой папки создать еще одну под названием **PC**. Сделав это, можете распаковать скачанный файл и изнутри директории **Hard Disk** скопировать папку **Software** и два распакованных файла **.hdf** в **Documents/FS-UAE/Hard rives/ClassicWB**.



4 Настройте эмулятор

Скопировав два распакованных файла, запустите эмулятор Amiga и измените модель Amiga на Amiga 1200. Затем перейдите во вкладку Hard Drives, щелкните по кнопке файла для первого диска и выберите распакованный файл **System_P96.hdf**. Подобным же образом нажмите на кнопки папок для следующих двух дисков и укажите папки **Software/** и **PC/** соответственно.



5 Установите ClassicWB

После этого вам нужно будет перейти во вкладку floppy и использовать кнопку + для добавления образов флоппи-диска workbench в список swap. Нажмите на кнопку Start, чтобы загрузить машину, и когда вам предложат вставить диск, нажмите F12 на клавиатуре, чтобы открыть меню FS-UAE. Просмотрите первый флоппи-диск и затем выберите диск workbench.



6 Завершите установку

Программа установки скопирует соответствующие файлы. Чтобы ускорить этот процесс, можно включить режим warp, нажав во время установки F12+W. По окончании процесса перейдите в главную вкладку, замените Config Name на **ClassicWB -- Amiga 1200** и нажмите кнопку Save. Теперь можете нажать на Start, чтобы загрузиться в ClassicWB. LXF

Терминал: Bash и Readline

Нейл Ботвик учит использовать оболочку, ничего не печатая... ну, почти ничего.



Наш эксперт

У **Нейла Ботвика** по компьютеру в каждой комнате, но по соображениям безопасности он нипочем не расскажет, где находится центральный сервер.

Начав использовать терминал, вы очень скоро поймете, насколько эффективнее делать все с клавиатуры вместо того, чтобы шаркать мышью по окнам и меню. Но — куда же без «но», не так ли? «Но» нашлось и у нас: вам понадобятся навыки весьма быстрой и точной печати. Если вы часто делаете очепатки, как я [вот и наглец, — *ред.*], то вам понадобится помощь.

В прошлом мы рассматривали всякие оперативные клавиши, подобные алиасам оболочки [см. Учебники, стр. 66, **LXF187**], которые позволяют определять горячие клавиши для заданных команд. Теперь мы рассмотрим еще парочку — макросы и функции. Оболочка *Bash* использует системный вызов *Readline* [чтения строки] для ввода и редактирования текста, что дает неплохие возможности индивидуальной настройки. Это делается с помощью файла **inputrc**, и наша пара тут доступна.

Введите inputrc

Как обычно, **/etc/inputrc** дает общие определения, а в **~/.inputrc** добавляются определения от отдельного пользователя. Если один и тот же макрос определен в обоих, то приоритет имеет определение пользователя. Файл **inputrc** являет собой обычный текст, однако он не читается при каждом вызове *Readline* — это было бы медленно. Он анализируется при открытии оболочки, и если вы внесли изменения, вам придется открыть оболочку заново, чтобы увидеть эффект.

› **В /etc/inputrc есть примеры определений клавиш, с комментариями.**

Скажем, вы часто переадресуете стандартный результат и стандартную ошибку некой команды в **/tmp/output.txt** (я предпочитаю использовать **/tmp**, чтобы не загромождать свою домашнюю директорию), но не хотите печатать это снова и снова. Откройте **~/.inputrc** в своем любимом текстовом редакторе, создайте его, если его не существует, и добавьте следующую строку:

```
Control-o: " &>|/tmp/output.txt"
```

Строка начинается с используемой комбинации клавиш — в данном случае это Ctrl+o; обратите внимание, что Control пишется полностью, а не просто Ctrl. За ним следует двоеточие (:) без пробела, *Readline* очень строг насчет формата этого файла. Затем макрос заключается в двойные кавычки — и кавычки обязательны. Теперь можете ввести имя команды и нажать Ctrl+o, чтобы добавить переадресацию. Если вы не знакомы с переадресацией, **>** отправляет результат команды в указанный файл, **&>** отправляет также сообщения об ошибках. Здесь присутствует |, поскольку в некоторых оболочках включена функция 'noclobber', которая не дает разрушить файл, перезаписав его с помощью **>**, а | позволяет переписать файл.

Клавиши указываются двумя способами. Человеко-читаемая опция — использовать английские названия: Control и Meta в качестве модификаторов, и название клавиши. У некоторых клавиш английские названия — DEL, ESC, ENTER, SPACE и TAB (обратите внимание на заглавные буквы).

В спецификации клавиши не обязательно должен быть один модификатор и одна клавиша: вы можете использовать несколько модификаторов (Control+Meta+u), или вообще не использовать модификаторов, хотя это принесет мало пользы, если только вы не хотите свести кого-то с ума, сбросив ему **.inputrc**, который по-новому определяет буквы, в директорию **home**. В некоторых ситуациях это бывает удобно для определения поведения небуквенных клавиш с помощью функций, о которых мы скоро поговорим.

Другой способ спецификации клавиш — с помощью их первичных кодов [raw codes], что не так наглядно при чтении, зато работает для всех клавиш. Конечно, вам понадобится отыскать соответствующую последовательность клавиш, однако у *Bash* для этого есть одна хитрость. Нажмите Ctrl+V и затем — комбинацию клавиш, которую хотите использовать. Например, Ctrl+F10 дает

```
^[[21;5~
```

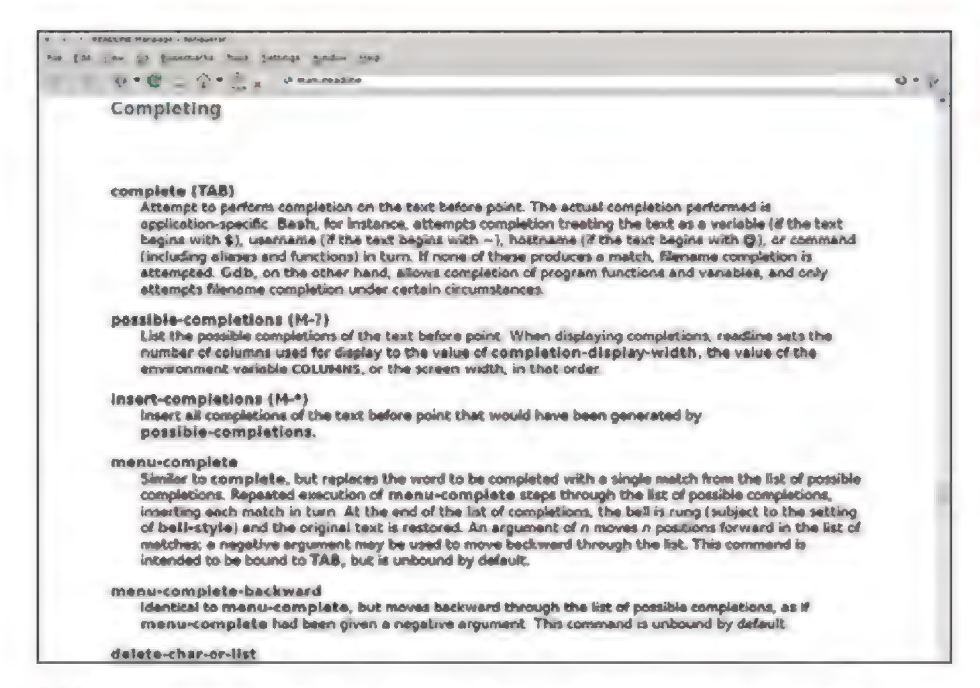
Открывающая **^[** — это экранирование [escape], которое в **.inputrc** представлено как **\e**, и если добавить в файл

```
# Ctrl-F10
"\e[21;5~": "Запускаем мировое господство"
```

то в терминале напечатается данный текст. Помните, что этот текст при подобной его спецификации нужно заключить в двойные

Горячие клавиши рабочего стола

От наличия удобных горячих клавиш выигрывает не только терминал, хотя **inputrc** работает только там. Основные среды рабочего стола имеют редакторы, позволяющие добавить или изменить клавиши быстрого запуска. Есть также *XBindKeys* (http://bit.ly/XBindKeys), не зависящая от рабочего стола программа для настойки клавиш быстрого запуска в X. Какой бы метод вы ни выбрали, удачно выбранные горячие клавиши могут сильно упростить вашу жизнь. А поскольку вы сами выбираете эти комбинации клавиш, вы должны без труда их запомнить.



› Man-страница *Readline* предлагает очень длинный список имеющихся функций, пригодных для задания поведения вашей оболочки.

кавычки. Желая избежать ломания головы, разумно было бы включить комментарии к добавленным таким образом клавишам, если только вы не уверены твердо, что с ходу вспомните значение **"\e[21;5~"**, когда решите взглянуть на файл через годик.

Печатать текст в терминале исключительно удобно и позволяет вам сократить набор текста (и повторный набор в случае ошибки), но это — только начало гениальности **inputrc**. Он также может выполнять функции, а в *Readline* есть немало встроенных. Man-страница Readline перечисляет функции, многие из которых могут быть вам знакомы и, вероятно, уже приписаны к определенным клавишам; но вы можете добавить новые или переделать привязку уже существующих. Если вы наберете часть имени файла в Bash и нажмете Tab, *Readline* попытается по максимуму дополнить его и далее покажет вам список возможных совпадений. Затем он будет ждать, что вы введете еще несколько символов и снова нажмете Tab, ибо это — функция автодополнения *Readline*. Есть альтернативная функция, menu-complete, которая в цикле проходит по возможным совпадениям при всех последующих нажатиях Tab, во многом подобно *Zsh*. Вы можете добавить это действие в *Bash*, добавив следующую строку в **inputrc**:

```
TAB: menu-complete
```

Еще одна функция, которая может показаться вам удобной во время экспериментов с настройками **inputrc** — это re-read-init-file; она делает именно то, что и заявляет (перечитывает файл init). Возможно, она даже окажется уже привязанной к последовательности клавиш Ctrl+X Ctrl+R — никто не запрещал последовательности клавиш быть не одной клавишей или обойтись без модификатора. Это эквивалент строки **inputrc**

```
"\C-x\C-r": re-read-init-file
```

К этому моменту применение кавычек уже может запутать. Поясним. Слева — определение клавиши; строка без кавычек, если вы используете обычные определения на английском, и в кавычках при использовании первичных определений. Справа — она заключена в кавычки, если это макрос, и не заключена в кавычки, если это имя функции. Усвоили?

Функции

Можно определить для *Readline* новые функции, но их пишут на C, и они не являются целью нашего урока. Если вам нужна последовательность команд для запуска в оболочке, возможно, правильнее будет определить функцию оболочки. Это делается точно так же, как определение функции в скрипте оболочки, разве что вы делаете это в файле вашего профиля, **~/.bash_profile** или **~/.bashrc**, в зависимости от того, используете ли вы оболочку логина. Вот обычный пример:

```
count() {
  for n in {1..10} ; do
   echo $n
  done
}
```

Перезапустите *Bash* и **введите** count, и напечатаются цифры от одного до десяти. Теперь можете привязать их к клавише с помощью

```
Control-t: "count\n"
```

Мы заканчиваем макрос сочетанием **\n**, чтобы имя функции печаталось с последующим Enter. Как мы и сказали, это тривиально, но показывает потенциал ваших файлов **profile** и **inputrc**: вы можете определить почти любую последовательность команд, какую хотите, и привязать ее почти к любой комбинации клавиш по своему усмотрению.

Другие оболочки

Настройки *Readline* соответствуют тому экземпляру оболочки, который работает. Если вы войдете от имени другого пользователя, у вас может быть другая настройка. Стоит также помнить, что если при SSH на другой компьютер вы используете оболочку на том компьютере, и ваш локальный **.inputrc** более не актуален. Вы можете заметить, что здесь мы ссылаемся исключительно на *Bash*: не все оболочки используют *Readline*. В частности, *Zsh* не использует. Это не значит, что вы не можете определить клавиши в *Zsh*, но это делается не через **inputrc**. Вместо этого вы используете команду *bindkey* в своем **.zshrc**. Чтобы клавиша или последовательность клавиш давала другой результат, используйте **bindkey -s**, поэтому эквивалентом приведенного выше примера с Ctrl+F10 будет:

```
bindkey -s "\e[21;5~" "Запускаем мировое господство"
```

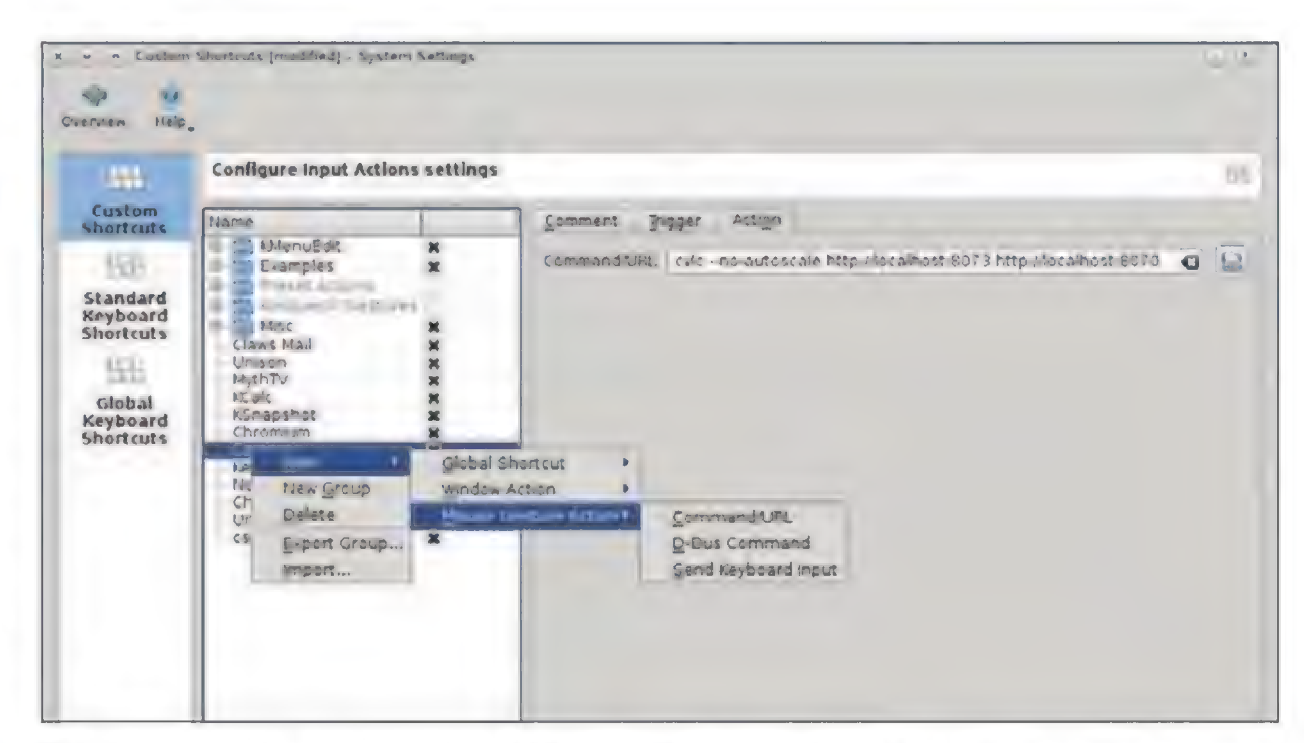
Bindkey можно запустить в терминале, так что вы сможете проверить подстановки в текущей сессии, прежде чем передавать их в **.zshrc**. LXF

Зовись она иначе...

Раньше мы уже рассматривали алиасы (псевдонимы) оболочки, но это очень полезное дополнение, и не грех скоренько освежить память о них. Команда алиаса выглядит примерно так:

```
alias l='ls -lhA --color=auto'
```

Поэтому, когда я ввожу в терминале **l**, запускается **ls** с данными опциями, позволяя избежать необходимости вводить что-то или что-то запоминать. Алиас не включает Enter, так что добавляйте при использовании. **l** выводит список текущей директории, а **l -R Documents** дает рекурсивный список директории **Documents**. Возможно, у вас уже настроены какие-то алиасы; запустите команду **alias** без аргументов, чтобы вывести их перечень. Команды алиасов обычно добавляются в ваш файл **profile**, поэтому они доступны во всех оболочках.



› Большинство рабочих столов предлагают редактор горячих клавиш — здесь показано предложение от KDE.

Nginx: Добавим страницы XHTML

Благополучно установив стек LAMP, **Эндрю Моллетт** объясняет, как создавать динамические сайты и размещать их на даже на устройствах вроде Raspberry Pi.



Наш эксперт

Эндрю Моллетт — преподаватель по Linux с более чем 700 видероликами на YouTube (http://bit.ly/UrbPeng). Его курсы также можно найти на www.pluralsight.com.



Ранее мы посмотрели, как установить и настроить web-сервер *Nginx*, но стартовую страницу, которую мы создали, было трудно назвать привлекательной для потенциальных посетителей. Наш сугубо текстовый сайт можно немного развить, обратившись к HTML (Hyper Text Markup Language — язык разметки гипертекста) и CSS (Cascading Style Sheets — каскадные таблицы стилей). На HTML мы создадим содержание сайта, а CSS поможет создать единое оформление для него. Далее мы обсудим HTML и CSS более подробно, а также уделим внимание ближайшему родственнику HTML, стандарту XHTML. Все это позволит вам проектировать и создавать web-страницы в одном стиле. Эти страницы будут отвечать стандартам создания web-страниц и будут максимально совместимы с web-браузерами. Аппаратная составляющая останется неизменной — мы воспользуемся Raspberry Pi с объемом ОЗУ 512 МБ, на котором стоит Debian Wheezy 7.5 с установленным и запущенным web-сервером *Nginx*. Мы настроили корневой каталог документа (Document Root) в *Nginx* (**/var/www**), и чтобы следовать за нами далее, у вас должно быть право создавать и изменять содержимое в этом каталоге.

Как мы упомянули ранее, HTML — сокращение от "Hyper Text Markup Language [язык разметки гипертекста]". Это стандарт документа, который позволяет web-клиентам, или браузерам, таким как *Firefox* или *Safari*, отображать текст, изображения и ссылки с некоторым форматированием, или «разметкой». Изначально HTML был предназначен для публикации научных статей. Ранний Интернет во многом был привязан к требованиям университетов, в которых и была разработана значительная часть сегодняшних стандартов. Возможность ссылаться на другие работы имела важнейшее значение — благодаря ей любой, кто читает документ, мог легко пройти по ссылке и прочесть оба документа. Многие называют отцом Интернета Тима Бернерса-Ли [Tim Berners-Lee], но на самом деле он разработал Всемирную сеть, которая позволяет пользоваться Интернетом. Интернет, позволивший компьютерам общаться друг с другом, был разработан в шестидесятых годах американскими военными и университетами. Сэр Тим Бернерс-Ли, выпускник Оксфорда, разработал то, что сейчас известно как Интернет, в 1990 году и определил спецификации HTML, URI (универсальный идентификатор ресурса, Universal Resource Identifier, который идентифицирует документы в сети) и HTTP (протокол передачи гипертекста, Hyper Text Transfer Protocol — язык, на котором браузер общается с web-сервером). По сей день сэр Тим остается директором W3C, консорциума Всемирной сети (World Wide Web Consortium), который поддерживает и разрабатывает стандарты web-протоколов.

Объясняем HTML

HTML используется для аннотации текстовых документов, из которых состоят web-страницы. Web-страница представляет собой простой текст, но с помощью аннотаций HTML-кода текст можно сделать жирным или превратить в гиперссылку. С помощью «тэгов» мы можем определять элементы страницы, которые будут интерпретироваться браузером определенным образом. У каждого элемента есть открывающий и закрывающий тэги, которые определяют начало и конец этого элемента. Стандарт HTML, в соответствии с которым мы создаем web-страницы, определяет, какие закрывающие тэги должны использоваться, и далее мы ближе познакомимся с ним и с другими стандартами, которые могут нам пригодиться.

Web-страницы обычно имеют расширение ".html", а стартовая страница сайта обычно называется "index.html". Эта страница открывается по умолчанию, если в URI не указана конкретная страница. Например, в адресе www.example.com конкретный документ не указан, поэтому будет открыта страница index.html; а если бы мы указали URI www.example.com/about.html, браузеру была бы возвращена указанная страница.

Страницу index.html можно создать в любом текстовом редакторе. Однако в зависимости от того, насколько сложен ваш документ, может потребоваться специализированный редактор. В Raspberry Pi можно воспользоваться *nano* или даже *vi* для

Валидация HTML

Любой HTML- или XHTML-код, который вы планируете использовать для сайта, легко проверить на соответствие требованиям W3C с помощью валидатора на официальном сайте консорциума: http://validator.w3.org. Код можно вставить в валидатор, загрузить в виде HTML-файла или указать адрес уже размещенной в Интернете страницы. В любом коде, который успешно прошел проверку, можно разместить логотип W3C со стандартами, которым соответствует ваша страница. Валидатор также может помочь в изучении синтаксиса, так как для обнаруженных ошибок выводятся пояснения. Наконец, написание кода в соответствии со стандартом обеспечит одинаковый внешний вид страницы в самых разных браузерах.

создания и изменения документа с командной строки и *LeafPad* — для того же самого в графической программе. Также можем посоветовать *Geany*, текстовый редактор и открытую IDE (Integrated Design Environment — интегрированная среда разработки), в которой удобно писать код на HTML и на других языках, таких как C или Python. Подсветка синтаксиса сделает код более читаемым при редактировании, а в *Geany*, например, при создании открывающего тэга автоматически вставляется закрывающий тэг. Для установки *Geany* на Raspberry Pi воспользуйтесь командами:

```
sudo apt-get update
sudo apt-get install geany
```

This document was successfully checked as XHTML 1.0 Strict!		
Result:	Passed	
Address :	http://theurbanpenguin.com/	
Encoding :	utf-8	(detect automatically)
Doctype :	XHTML 1.0 Strict	(detect automatically)
Root Element:	html	
Root Namespace:	http://www.w3.org/1999/xhtml	

› **Для проверки соответствия кода HTML-стандартам можно воспользоваться валидатором на www.W3.org.**

Разбираем тэги

Как мы видели на первом уроке, web-страница может быть чисто текстовой, без тэгов HTML. В этом случае отображается текст без разметки. Как и наша текущая страница, такая страница будет смотреться пресно, не привлечет читателей, и на ней нельзя будет выделить информацию. Мы повысим интерес к ней и улучшим читаемость с помощью тэгов HTML; и начнем с DOCTYPE.

» <DOCTYPE> Этот информационный тэг, который не является частью разметки документа. Он используется для указания стандарта HTML, в соответствии с которым написан документ и в соответствии с которым браузер будет интерпретировать его. Данный тэг можно опустить, но это рискованно, потому что браузеру придется догадываться о том, для какого стандарта написана страница. Просить программу о чем-то догадаться — всегда плохая идея. Включение этого тэга также позволит нам провести валидацию кода в соответствии со стандартом и убедиться, что мы не забыли указать никакие атрибуты или закрывающие тэги. Проверить свой код можно, зайдя на http://validator.w3.org. На нашем уроке мы будем использовать тип XHTML, и тэг для него имеет следующий вид:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Самые зоркие из вас заметят, что с помощью XML мы также задали кодировку по умолчанию, в данном случае, UTF-8. Кодировка — часть стандарта XHTML, и ее можно задать в документе или в настройках сервера *Nginx*: "charset utf-8;". Зачем использовать XHTML поверх HTML? XHTML — сокращение от "eXtensible Hyper Text Markup Language [Расширенный язык разметки гипертекста]", и он почти идентичен HTML 4.01, за исключением большей строгости и чистоты стандарта. XHTML переопределяет HTML как XML-приложение с поддержкой во всех крупных браузерах. XHTML предлагает лучший для разработчиков и браузеров стандарт, потому что некоторые необязательные элементы HTML в нем стали обязательными — к ним относятся тэги <html>, <head>, <title> и <body>. Тэги нужно писать в нижнем регистре, что, впрочем, не относится к с самому тэгу DOCTYPE. Для удобства чтения кода у каждого элемента должен быть соответствующий закрывающий тэг. В HTML при указании начального тэга абзаца <p> предыдущий такой тэг автоматически закрывается, если не был закрыт явно. Правильнее закрывать его явно тэгом </p>. Требование обязательно закрывать тэги позволяет избежать проблем, связанных с небрежным программированием. Некоторые открывающие тэги одновременно могут являться и закрывающими: к ним относятся горизонтальная линия <hr> или новая строка
. Чтобы тэг стал самозакрывающимся, его нужно записать, например, так:
. Эта корректная форма открывания и закрывания тэга в соответствии со стандартом XHTML. Так как XHTML является текущим стандартом HTML от W3C, мы реализуем его на своем сайте. Работа над HTML 5 продолжается — на данный момент это стандарт скорее черновой, но тем не менее многие браузеры его поддерживают.

» <html> Тэг <html> задает корень HTML-документа и в XHTML является обязательным; также обязательным является его атрибут xmlns, который задает используемое в документе пространство имен XML. Тэг <html> выступает в качестве контейнера для всех остальных элементов страницы. Пример правильного определения тэга <html> в XHTML: <html xmlns="http://www.w3.org/1999/xhtml">. Как мы упоминали ранее, все элементы, включая сам тэг, должны записываться в нижнем регистре, а значения атрибутов должны заключаться в кавычки.

» <head> Тэг <head> — это контейнер, содержащий метаданные, которые описывают документ HTML. Большая их часть невидима пользователю; самое заметное исключение — элемент <title>, который задает название страницы, отображаемое на вкладке браузера. Наряду с <title>, тэг <head> может содержать элементы <meta>, которые предоставляют информацию о странице поисковым машинам, элементы <link> для включения таблиц стилей или <style> для определения стилей внутри самой страницы, и, наконец, элемент <script>, определяющий скрипты, которые могут запускаться на странице. Пример элемента <head> приведен ниже:

```
<head>
  <title>Linux Format HTML Fundamantala</title>
  <link rel="stylesheet" href="/stylesheets/screen.css" type="text/css" />
  <meta name="Author" content="Andrew Mallett" />
```

»

```
  <meta name="Description" content="HTML Fundamentals
Tutorial" />
  <script type="text/javascript">
    window.onload = function() {
      alert("The Pi is fun");
    };
  </script>
</head>
```

» <body> Элемент <body> выступает как контейнер для элементов оформления страницы, которые будут выводиться на экран браузером. Большую часть времени при работе с web-страницами вы проведете внутри элемента <body>. Первым тэгом в <body> часто является элемент заголовка <h1>. Тэги с <h1> по <h6> соответствуют заголовкам различного уровня. Тэг <h1> в документе может быть только один; это главный элемент, используемый для определения темы страницы при поиске. В теле страницы можно использовать другие блочные элементы, например, <div>, для разделения страницы на логические блоки, которыми допускается управлять отдельно.

На каждой web-странице есть один элемент <body>, но он может содержать несколько элементов <div>. С помощью атрибутов элементов и стилей CSS можно задать цвет, шрифт и размер текста, а также нужным образом расположить элементы на web-странице. У каждого элемента есть атрибуты ID и CLASS. ID используется для уникальной идентификации элемента на странице, а CLASS — для группировки элементов, которые должны быть оформлены в одном стиле. Если нам нужен вводный блок текста перед основным текстом, код мог бы был быть таким:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Linux Format</title>
    <link rel="stylesheet" href="stylesheets/screen.css"
type="text/css" />
    <meta name="Author" content="Эндрю Моллетт" />
    <meta name="Description" content="Учебник по HTML" />
    <script type="text/javascript">
      window.onload = function() {
        alert("Pi – это забавно");
      };
    </script>
    <style type="text/css">
      body {color:red}
      #overview {color:darkslategray; font-size:x-large}
      #overview::first-letter {color:red; font-size:xx-large}
      #content {color:lightslategray; font-size:medium}
      .tutorial {text-indent:50px}
    </style>
  </head>
    <body>
      <h1>HTML Fundamentals</h1>
        <div id="overview" class="tutorial">
          <p>Это урок по HTML и применению тэгов</p>
        </div>
        <div id="content" class="tutorial">
          <p>HTML разрабтал Тим Бернерс-Ли. <br />
            Тим закончил Оксфордский университет.
          </p>
        </div>
      </body>
</html>
```

Вложенный элемент <style> задает красный цвет текста в <body>; при этом элемент с идентификатором "overview" будет цвета темного шифера (darkslategray), а элемент с идентификатором "content" будет более светлого оттенка серого. Стили также влияют на размер шрифта. Чтобы сделать все чуть поинтереснее, сделаем у первых строк каждого элемента класса "tutorial" отступ в 50 пикселей, а первую букву элемента с идентификатором "overview" сделаем красной и чуть большей по размеру, чем основной текст. Для обращения к идентификаторам в CSS используется #, а точка означает обращение к классу элементов. Как видно из использования "body" во включенном в страницу CSS, к элементам HTML можно обращаться по тэгам.

» Пробелы Текст на странице в HTML может разделяться одиночными пробелами; любые дополнительные пробелы, табуляция и переводы строки будут проигнорированы. Между словами в документе можно указать несколько пробелов, но в результате на странице появится только один пробел, а лишние новые строки во внимание не примут. Для добавления новой строки можно воспользоваться самозакрывающимся тэгом
 или указать <p> в начале нового абзаца и затем добавить две новых строки. Каждый добавляемый элемент <div> также начнется с новой строки. <div> может содержать один или несколько элементов <p>, но абзац <p> по стандарту XHTML не может включать <div>.

CSS3

Мы уже немного познакомились с таблицами стилей при разработке нашей XHTML-страницы; однако при включении таблицы стилей в элемент <head> они не определяют все страницы автоматически. Хотя информацию о стилях можно было бы скопировать и вставить в каждую страницу, в дальнейшем возникнут сложности, так как изменения придется делать в каждой странице. Чтобы по-настоящему воспользоваться таблицами стилей, мы включим в страницу в <head> элемент <link>, ссылающийся на внешнюю таблицу стилей, которая будет обслуживать все страницы. Мы будем пользоваться CSS3, ядро которого основано на CSS2.1, с дополнительными модулями.

Сначала создадим страницу XHTML, которая будет основой нашего проекта, после чего добавим информацию о стилях в эту и другие страницы:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Linux Format</title> <meta name="Author" content="Эндрю Моллетт" />
<meta name="Description" content="Учебник по HTML" />
</head>
  <body>
```

HTML Fundamentals

- Home
- Search

Welcome the Linux Format's HTML tutorials with CSS3

› Наша старая добрая страница XHTML — стартовая точка для добавления стилей.

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

```
<div id="page">
<div id="header" class="frame">
<h1>HTML Fundamentals</h1>
</div>
<div id="menu" class="frame">
<ul>
<li><a id="home" href= "/index.html">Home</a></li>
<li><a id="search" href= "/search.html">Search</a></li>
</ul>
</div>
<div id="content" class="frame">
Welcome the Linux Format's HTML tutorials with CSS3
</div>
</div>
  </body>
</html>
```

Полученную страницу не назовешь шедевром, но у нас есть немного чистого текста, к которому можно применить стили. Для простоты создадим таблицу стилей в том же каталоге, где находится индексная страница, **/var/www** на Raspberry Pi. Для этого в *Geany* или в другом текстовом редакторе создайте файл **/var/www/style.css**. Для ссылки на таблицу стилей из индексной страницы и других страниц, которые должны иметь такой же стиль — например, страница поиска — достаточно включить следующую строку в элемент <head>:

```
<link rel="stylesheet" href="/style.css" />
```

CSS-файл начинается с указания используемой кодировки:

```
@charset 'utf-8';
```

Взглянув на код XHTML, вы увидите элемент <div> с идентификатором страницы, который является главным контейнером для всех остальных элементов. Теперь применим к нему стиль:

```
#page { width: 700px; margin: auto; border: solid 1px darkslate-
gray; border-radius: 5px; box-shadow: 4px 4px 4 px black; }
```

При определении стилей у каждого стиля указываются пары «имя параметра: значение», разделенные двоеточиями (:), каждая пара завершается точкой с запятой (;). Для лучшей читаемости атрибуты можно разбить на несколько строк. Добавив стиль и обновив страницу в браузере, вы увидите, что страница выглядит чуть интереснее — она выровнена по центру и заключена в контур с закругленными углами и затенением.

Следующий элемент нашей XHTML-страницы — заголовок с идентификатором "header". Это главный баннер, и мы зададим его фоновый цвет и высоту. Добавьте следующий код в CSS, чтобы изменить только элемент с идентификатором "header":

```
#header { background-color: slategray; color: white; height: 55px; }
```

Мы включили класс "frame" в каждый элемент <div> внутри главного контейнера <div>. С помощью этого класса можно управлять стилями, которые должны быть постоянными у нескольких элементов на странице:

```
.frame { width: 700px; margin: 0px; padding 0px; background-
color: white; }
```

Так как мы задаем цвет фона **background-color** и через идентификатор, и через класс, для заголовка будет более приоритетным стиль идентификатора. Для форматирования элемента <h1> нашего заголовка можно воспользоваться следующим стилем, который повлияет только на элементы <h1> — если помните, в документе может быть только один заголовок <h1>:

```
h1 { text-align: center; text-shadow: 4px 4px 4px black; fontsize:
300%; }
```

Следующий элемент страницы — контейнер <div> с идентификатором "menu". Он содержит ссылки для навигации по нашему сайту. Сами ссылки содержатся внутри контейнера в виде неупорядоченного списка, и всем этим элементам нужно придать некоторый стиль:

```
#menu {
  font-family: sans-serif; background-color: black;
  height: 22px; line-height: 22px; position: relative;
  }
#menu ul {
  list-style-type: none; margin: 0; padding: 0;
  }
#menu li {
  display: inline; float: left; position: relative; padding: 0 5px;
  }
#menu a:link, #menu a:visited {
  background-color: black; color: white;
  display: block; font-weight: normal;
  padding: 0 5px; text-decoration: none;
}
#menu li a:hover, #menu li a:active,
#menu li.selected a:link, #menu li.selected a:visited {
  background-color: white; color: black;
  }
```

Здесь много форматирования: мы устанавливаем фоновый цвет верхней панели меню в черный, а текста на этой панели — в белый; ссылки отображаем горизонтально, и при наведении курсора мыши на меню реверсируем цвета. Эффект довольно приятный, и чтобы добиться его, не нужно много усилий. Наконец, мы задаем стиль элемента <div> с содержимым страницы:

```
#content { padding: 25px 50px; width: 600px; font-size: 150%; }
```

Внешние поля (края) задаются в 25 пикселей вверху и внизу и в 50 пикселей слева и справа. Общая ширина элемента устанавливается в 600 пикселей, так как общая ширина страницы, которая есть у нас в распоряжении — 700 пикселей. Теперь наш сайт выглядит лучше, и хотя поисковой страницы еще нет, легко создать новые страницы, воспользовавшись индексной в качестве шаблона. Стили легко применить к новым страницам.

Страницу поиска еще нужно написать, но для этого достаточно скопировать индексную страницу и изменить элементы <title> и <h1>. Подробнее об этом — через месяц, когда мы посмотрим, как добавить в нашу страницу код на PHP.

› Наша страница с добавленными стилями из файла CSS, созданного нами в /var/www.

Поля и края CSS

При применении полей [padding] и краев [margin] к элементам нужно понимать, что поля находятся вне элементов, а края — внутри элементов. У каждого из этих элементов есть четыре атрибута: верхнее [top], правое [right], нижнее [bottom] и левое [left] значения. Если указаны все четыре значения, они считываются в этом порядке. Если указаны только два значения, то первому значению соответствуют верхнее и нижнее поля или края, а второму — левое и правое поля или края. Если указано только одно значение, то оно приписывается всем четырем атрибутам.

Макросы: Офис и автоматизация

Хотите до ужаса быстро работать в бесплатном офисном редакторе? Следуйте за **Марко Фиоретти** в рассказе о макросах *LibreOffice* (и *OpenOffice*).



Наш эксперт

Марко Фиоретти любит свободное ПО. Он изучает и преподает влияние открытых цифровых технологий на общество.

Все, кто проводят большую часть времени за офисными документами — написанием или изменением текста, работой с электронными таблицами или презентациями — должны знать о макросах. Они помогают сэкономить массу времени, а иногда и избежать неприятностей.

На нашем уроке мы покажем, как работать с макросами в *LibreOffice*, самой популярной программе среди начинающих линуксоидов. Этот редактор уже установлен в большинстве дистрибутивов для новичков; большая часть того, о чем мы будем говорить, также справедливо и для *OpenOffice*.

Сначала объясним, что такое офисные макросы и когда и зачем стоит ими пользоваться (или не стоит!). Потом очень быстро представим основные возможности языка макросов *LibreOffice* по умолчанию, а затем рассмотрим рабочий и, надеемся, очень полезный пример. Наконец, мы расскажем, как создать, запустить и написать макрос и как правильно хранить макросы.

Что такое макросы?

Насколько нам известно, макрос — это последовательность команд для программы (в данном случае, для *LibreOffice*), написанная и упакованная так, что ее можно быстро выполнить, просто запустив, как мы вскоре и увидим. Макросы способны существенно увеличить скорость выполнения частых, повторяющихся и относительно простых задач. Однако вредоносные или плохо написанные макросы могут принести вам самые разнообразные проблемы.

Макросы не нужно путать с расширениями. Это важно помнить потому, что *LibreOffice* здесь не так удобен, как хотелось бы. Например, на некоторых панелях (включая панель Object Dialog [Диалог объекта] в Organizer [Органайзер]) макросы и расширения находятся рядом друг с другом.

Расширение — это дополнительный программный пакет, у которого есть собственный установочный пакет и менеджер (Tools > Extension Manager [Инструменты > Менеджер расширений]). Обычно расширения работают не так, как макросы — на другом уровне.

Расширения могут быть написаны на любом языке программирования и обычно выполняют не повторяющиеся, довольно сложные задачи. В определенном смысле расширения — это способ добавления в *LibreOffice/OpenOffice* возможностей на базе (существующего) кода, которые невозможно или нецелесообразно писать или переписывать на Basic. К примерам возможностей, которые можно и нужно реализовывать в виде расширений, относятся, например, панель построения трехмерных графиков или сводных таблиц.

Внутри макроса можно использовать почти любые команды — из тех, что обычно генерируются при нажатии кнопок и выборе пунктов меню, путем вызова длинных функций, написанных с нуля.

Пользуйтесь макросами ответственно

Освоив макросы, вы обнаружите, что писать макросы и пользоваться ими (по крайней мере, некоторые виды макросов — см. врезку «Переносимы ли макросы?») настолько увлекательно, что последствия могут быть непредсказуемыми. Два малоизвестных факта о миграции Большого Мюнхена в *OpenOffice* — в том, что на миграцию 21 000 шаблонов и макросов было потрачено огромное количество времени и денег, а 40% из них оказались излишними или не нужными с самого начала (http://bit.ly/1qsLiwP).

Простой пример такой избыточности — все макросы, которые просто добавляют какой-то один и тот же короткий текст в точку положения курсора или неправильно форматируют абзацы (с длинными последовательностями выравнивания по центру, жирного текста, текста, выделенного цветом, и т. д.). Подходящее средство для решения первой задачи — автотекст [AutoText], а не макрос. Абзацы почти во всех случаях нужно форматировать с помощью стилей, а не заданием отдельных параметров текста вручную или с помощью макросов. Зачем отказываться от возможностей автотекста и стилей своей офисной программы ради чего-то, почти наверняка гораздо менее надежного или менее переносимого в долгосрочной перспективе?

Не поймите меня неправильно: макросы — отличная штука, иначе мы не писали бы эту статью; но дважды подумайте, прежде чем писать новые макросы, особенно если пользоваться ими будут и другие люди.

Before	After
Bill Gates is the greatest prog[Header (Default Style)] history of software.	Linus Torvalds is the greatest [Header (Default Style)] the history of software.
His operating system, Windows, is the safest, fastest and coolest ever!	His operating system, Linux, is the safest, fastest and coolest ever!
There is nothing that makes me feel better than running Windows! A thousand thanks, Bill Gates!	There is nothing that makes me feel better than running Linux! A thousand thanks, Linus Torvalds!
When it comes to office file formats, instead, the absolute best is the one called OpenXML	When it comes to office file formats, instead, the absolute best is the one called OpenDocument Format
Only using OpenXML on Windows I am sure that my documents will remain available and easy to process!	Only using OpenDocument Format on Linux I am sure that my documents will remain available and easy to process!

› Рис. 1. Один и тот же текст до (слева) и после (справа) запуска легко настраиваемого макроса множественного поиска и замены, описанного на этих страницах.

Поговорим на языке макросов

По умолчанию макросы в *LibreOffice* пишутся на языке под названием Basic (оригинально, не правда ли?). Первые две вещи, которые нужно знать об этом языке — ключевые слова в нем не чувствительны к регистру, и все, что находится в строке после "REM" или заключено в одиночные кавычки, представляет собой комментарии.

Некоторые команды Basic так просты, что их смысл очевиден, даже если у вас нет ни малейшего знакомства с другими языками программирования (а у вас, как у рьяного читателя Linux Format, оно определенно есть, не так ли?). Например, это циклы 'for' и 'while'. Другие команды, например, некоторые структуры данных, имеют такие длинные и странные имена, что вы никогда бы не догадались об их существовании. К счастью, найдя их в справочных материалах, вы увидите, что они не так сложны, как кажется на первый взгляд.

Не беспокойтесь о них. Для экономии места мы расскажем лишь о самых простых возможностях Basic в *LibreOffice*, который почти идентичен своему собрату из *OpenOffice*.

Переменные и массивы определяются с помощью ключевого слова DIM, которое обязательно лишь во втором из указанных случаев:

```
Const Pi=3.14
Dim Names$(5,4)
```

Первая строка выше — объявление константы. Во второй объявляется матрица из 30 строк, разделенных на шесть столбцов (с номерами от 0 до 5) и пять строк (с номерами от -2 до 2). Как вы видите, индексы массивов могут быть отрицательными. Знак доллара означает, что это массив строк. Целочисленные значения обозначаются аналогично, знаком %, а валюты — знаком @. Объявлять типы можно также явно:

```
Dim Price@
Dim Age%
Dim Price As Currency
Dim Age as Integer
```

Первые два выражения эквивалентны последним двум. Выражение 'Option Explicit' в начале макроса означает, что все его переменные нужно декларировать. Да, это не обязательно, но никогда не забывайте об этом. Это поможет вам на раннем этапе найти гораздо больше ошибок, чем вы могли бы подумать.

Basic *LibreOffice* поддерживает множество типов текстовых, логических и числовых переменных (включая даты), а также массивы. Логические переменные — это переменные, которые могут принимать только значения False [ложно] или True [истинно]. При присваивании значения логической переменной 0 (ноль) означает False, а любое другое значение означает True. Также есть объекты [Objects] и общие сложные структуры, такие как таблицы или свойства — то есть пары имя — значение. Ниже вы как раз видите одно из тех эзотерических названий, о которых мы говорили ранее:

```
dim magazine(0) as new com.sun.star.beans.PropertyValue
magazine(0).Name = «Самый отличный журнал»
magazine(0).Value = «Linux Format»
```



› Органайзер в *LibreOffice* — центральная панель управления для редактирования, отладки одиночных макросов и целых библиотек и управления ими.

Чтобы узнать об этом подробнее, проследуйте по ссылкам, указанным во врезке «Источники и ресурсы», стр. 83.

Подпрограммы и функции

С технической точки зрения отдельные макросы представляют собой не что иное, как подпрограммы или функции Basic. И то, и другое — блоки повторно используемого кода, которые обычно делают что-то одно, но могут вызывать другие макросы. Функции возвращают какое-то значение, производя и передавая его обратно в код, который их вызвал. Подпрограммы значения не возвращают. Следовательно, в отличие от подпрограмм функции можно помещать в любое место кода, куда можно поместить значение того же типа, который возвращает функция (например, вызов функции можно присвоить переменной), но различия между подпрограммами и функциями на том и заканчиваются. Вот так объявляются и вызываются функции и подпрограммы соответственно:

```
Function Sum_Of_Two_Integers(FirstInt As Integer, SecondInt as Integer)
... ЗДЕСЬ ИДЕТ КОД ФУНКЦИИ
End Function
Total=Sum_Of_Two_Integers(SomeNumber; 58)
REM а вот пошла подпрограмма:
Sub Print_Readability
... ЗДЕСЬ ИДЕТ КОД ПОДПРОГРАММЫ
End Sub
```

Скорая помощь

Никогда не пользуйтесь макросами вместо стилей для получения постоянного, не зависящего от содержания форматирования. С помощью макросов можно применять стили автоматически, но не обходиться без стилей. Если вы забудете это золотое правило, то однажды пожалеете.

Диспетчеризация и команды

Макросы взаимодействуют с различными компонентами *LibreOffice* (от элементов пользовательского интерфейса, таких как панели инструментов, до внутренних процедур, работающих с открытыми файлами) посредством механизма, известного как »

Переносимы ли макросы?

Отличный вопрос. Это зависит от того, что делают макросы и где они должны это делать — только на всех ваших компьютерах или вообще везде?

Файлы OpenDocument хорошо переносимы, так как представляют собой документы, но макросы представляют собой код и должны взаимодействовать с офисным редактором более тесно, чем редактируемые в нем документы. Макросы, предназначенные и используемые для добавления каких-то возможностей в заданную программу или для автоматизации некоторых задач при работе с документами, прекрасно переносятся в тех рамках, о которых вы только что прочли в этом предложении. Макрос в приведенном примере должен работать точно так же во всех последних версиях *LibreOffice*, в любой операционной системе, сейчас и в обозримом будущем. Скорее всего, он также будет работать (в краткосрочном и среднесрочном периодах) в *OpenOffice*, так как эти два редактора довольно близки. Эта совместимость со временем склонна уменьшаться.

Но даже сегодня макросы на Basic скорее всего не будут работать ни в одном другом офисном редакторе, включая свободные вроде *Calligra*, которые устроены по-другому. Поэтому нужно дважды подумать, прежде чем передавать кому-то документы с макросами внутри!

диспетчеризация. Это также называется действиями. Например, при нажатии Ctrl+V и при выборе Edit > Paste генерируется одна и та же команда диспетчеризации. Вот сокращенный пример того, как может выглядеть диспетчеризация:

```
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.
DispatchHelper")
dispatcher.executeDispatch(document, ".uno:InsertText", "", 0,
SomeVariable)
```

Первая строка привязывает к объекту под названием 'document' текущий активный сервис *LibreOffice* [CurrentController] на момент вызова — в данном случае это главный фрейм [Frame] активного документа [ThisComponent].

Во второй строке кода создается диспетчер сообщений, используемый в третьей строке для оповещения CurrentController о том, что ему нужно вставить содержимое SomeVariable в определенное место. Диспетчер уродлив, но понять его несложно.

Перечислить все сервисы и типы объектов в Basic *LibreOffice* здесь невозможно. Просто чтобы дать вам представление, сообщим, что в таблицах *Calc* можно напрямую получить содержимое каждой ячейки методами getValue() и getString(). Что касается текста, есть масса способов поиска, написания, разбора и обработки текста. В следующем абзаце мы продемонстрируем один из них.

Скорая помощь

Утилитой записи макросов пренебрегать не стоит. Ее возможности в качестве генератора кода переоценены, и с ней макросы не могут принимать параметры, поэтому она не избавит вас от необходимости изучения Basic. Но как средство изучения Basic она может быть очень полезной.

Рабочий пример

Один из случаев, когда с помощью макросов очень удобно экономить время — постоянные многочисленные повторения одних и тех же наборов операций поиска и замены. Содержимое документа слева на рис. 1 (на стр. 80) было преобразовано в то, что вы видите справа, простым щелчком на макросе GetRecordStraight, который представляет собой измененную версию макроса, найденного на http://bit.ly/1liqBko:

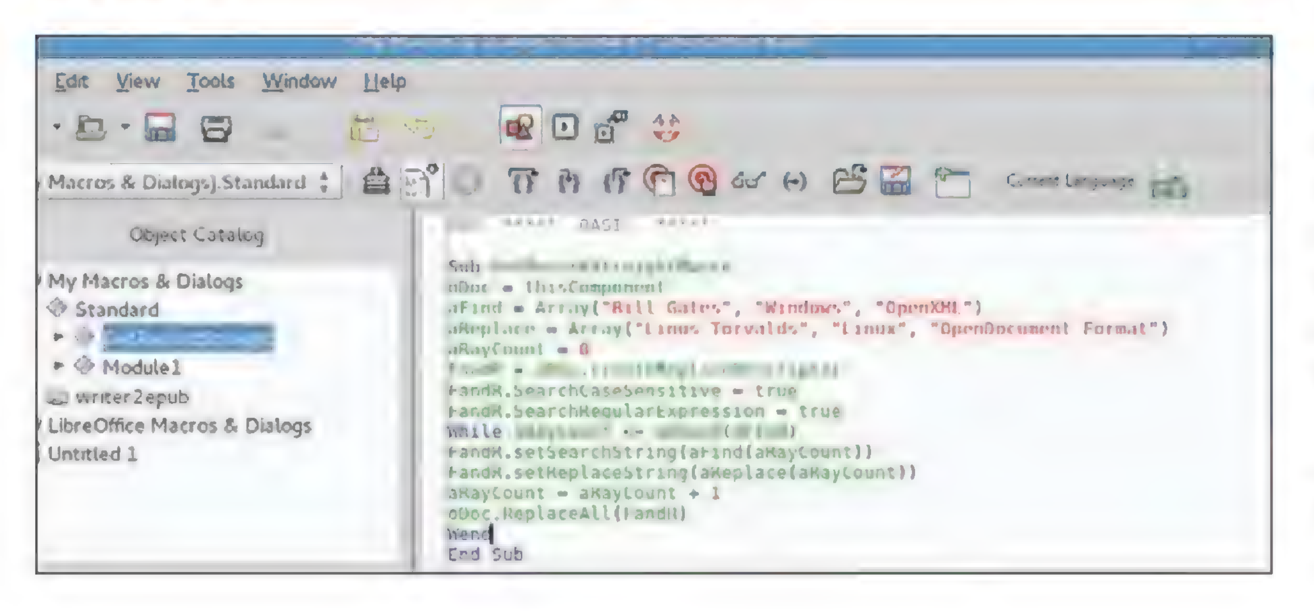
```
1 Sub GetRecordStraight
2 oDoc = thisComponent
3 aFind = Array("Bill Gates", "Windows", "OpenXML")
4 aReplace = Array("Linus Torvalds", "Linux", "OpenDocument
Format")
5 aRayCount = 0
6 FandR = oDoc.createReplaceDescriptor
7 FandR.SearchCaseSensitive = true
8 FandR.SearchRegularExpression = true
9 While aRayCount <= uBound(aFind)
10 FandR.setSearchString(aFind(aRayCount))
11 FandR.setReplaceString(aReplace(aRayCount))
12 aRayCount = aRayCount + 1
13 oDoc.ReplaceAll(FandR)
14 Wend
15 End Sub
```

Да, на первый взгляд этот код напоминает доисторического марсианина, но стоит взглянуть на него и снова на рис. 1, и вы мигом поймете, как он работает. Код начинается с двух массивов aFind и aReplace. Первый содержит все строки, которые вы хотите заменить в текущем документе (oDoc). Во втором массиве хранятся замены для этих строк, в том же порядке.

Единственный пример черной магии в этом макросе — объявление FandR в строке 6, но этой черной магией можно пользоваться без страха, если вы ее понимаете: createReplaceDescriptor — это объект с такими свойствами, как в строках 7 и 8 и 10 и 11, которые используются для операций поиска и замены. Например, для поиска без учета регистра нужно установить SearchCaseSensitive в false.

В цикле в строках с 9 по 14 мы просто проходим во всем элементам aFind, и каждый раз *LibreOffice* велит FandR (строка 13) заменить все вхождения текущей строки из aFind (строка 10) строками из массива aReplace (строка 11), находящимися в тех же позициях. Удобно, правда? Особенно если учесть, как просто заполнить два массива любыми необходимыми строками для замены!

› Верхняя часть редактора макросов выглядит простецки, но она поможет вам быстро тестировать и отлаживать свой код.



Создаем макрос без написания кода

«Написать» простой макрос можно, не набрав (и даже не увидев) ни одной строки кода на Basic. Утилита записи макросов [Macro Recorder] *LibreOffice* сохранит все команды, вызываемые при наборе текста и нажатии кнопок, в виде кода на Basic. Это особенно полезно как средство изучения Basic, поскольку позволяет увидеть, какой код соответствует определенным действиям. Однако возможности этой утилиты для полноценной записи макросов довольно ограничены.

Не все передаваемые команды записываются в полной, пригодной для последующего использования форме, включая действия на панелях настройки и в других диалоговых окон. По той же причине записанные макросы не могут принимать аргументы и в некоторых случаях способны вести себя не так, как ожидалось. Поэтому перед записью макроса рекомендуем вам записать все действия на бумаге, проверить их правильность и попробовать выполнить их до записи макроса.

Когда будете готовы к записи, сделайте следующее. Во-первых, включите утилиту записи макросов в меню Tools > Options > LibreOffice > Advanced [Инструменты > Настройки > LibreOffice > Дополнительно], затем выберите Tools > Macros > Record Macro [Инструменты > Макросы > Начать запись макроса]. Откроется маленькое окошко, которое покажет, что макрос записывается. Закончив запись, щелкните в нем и сохраните макрос. Как это сделать, мы объясним через минуту.

Пишем макрос...

Теперь, когда вы увидели рабочий код, пора научиться писать и сохранять его. Когда вы будете готовы к тому, чтобы написать свой первый макрос или просто вставить какой-нибудь код, выберите Tools > Macros > Organize Macros > LibreOffice Basic [Инструменты > Макросы > Управление макросами > LibreOffice Basic] и затем щелкните на Organizer [Органайзер] и New [Создать], чтобы создать новый макрос. Дайте ему подходящее имя и нажмите Edit [Изменить], после чего откроется редактор и отладчик макросов *LibreOffice*.

Наряду с базовыми функциями любого редактора у этой утилиты есть кнопки для запуска текущего макроса, даже пошагового, с помощью точек прерывания. При такой отладке на нижней панели будет показано, как с каждой командой изменяются значения всех переменных. Указатель в исходном коде показывает, какая команда выполняется. Кроме того, если поместить курсор над какой-то переменной во время выполнения, ее содержимое будет показано во всплывающем окне.

...и запускаем его!

Сначала откройте окно, не требующее пояснений, выбрав Tools > Options > Security > Macro Security [Инструменты > Настройки

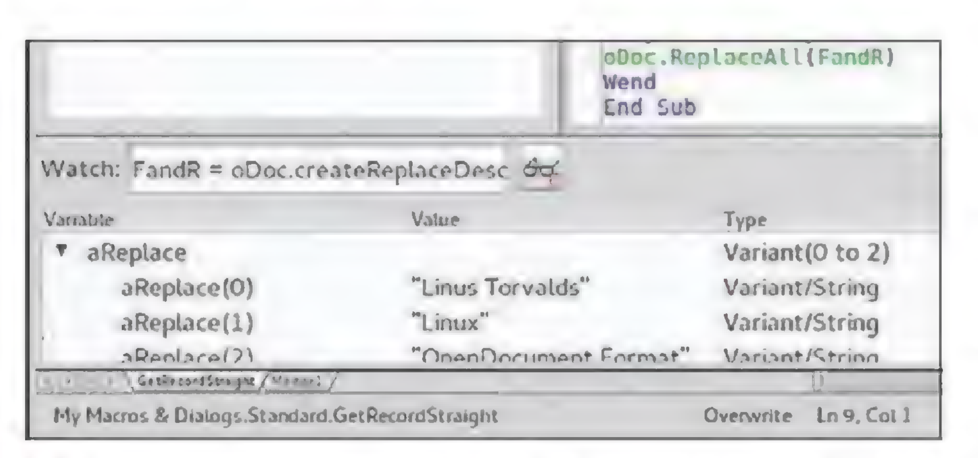
> Безопасность > Безопасность макросов]. Прочтите его содержимое очень внимательно и установите высочайший уровень безопасности, который вас устраивает. Макросы представляют собой исполняемый код. Если вы разрешите *LibreOffice* автоматически запускать макросы из любых источников и произойдет что-то плохое, виноваты будете вы сами!

Запускать макросы можно из Органайзера (см. выше) или из главного меню (Tools > Macros > Run Macro [Инструменты > Макросы > Запустить макрос]). В ячейках *Calc* макросы можно вызывать как обычные функции, даже несмотря на то, что их нет в списке функций *Calc*.

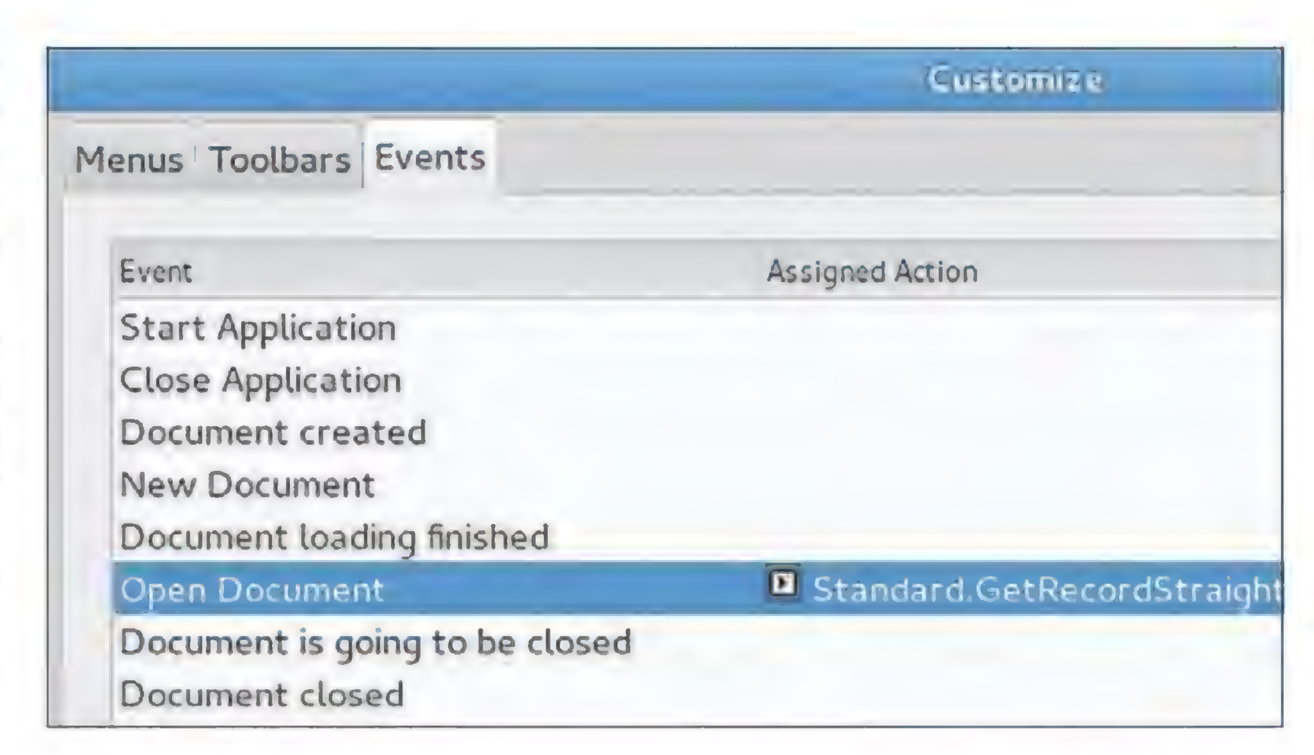
В принципе, самый эффективный способ запуска часто используемых макросов — привязать их к событиям или к компонентам пользовательского интерфейса. Нажмите кнопку Assign [Назначить] Органайзера или выберите Tools > Customize [Инструменты > Пользовательские настройки], и откроется окно с четырьмя вкладками по одной для каждой категории объектов: меню [Menus], клавиатура [Keyboard], панели инструментов [Toolbars] и события [Events]. Последняя вкладка содержит несколько десятков событий, таких как New Document или Document Title Changed. Выберите соответствующую вкладку и объект, к которому хотите привязать свой макрос. Затем нажмите Macro [Макрос] на вкладке Events [События] или выберите Add > Category > LibreOffice Macros [Добавить > Категория > Макрос LibreOffice] и выберите макрос. Если вы добавляете макрос на панель инструментов, придется перетащить его в нужное место и отпустить; в противном случае просто нажмите Add [Добавить].

Управление макросами

Как вы уже знаете, макросы представляют собой простые одиночные процедуры или функции. В *LibreOffice* объявления макросов и переменных нужно группировать — обычно по функциональности — в модули, размер которых ограничен 64 КБ. Несколько модулей, решающих аналогичные задачи, образуют библиотеку. Наконец, библиотеки хранятся в контейнерах. В *LibreOffice* есть два отдельных контейнера: для собственных макросов (под названием LibreOffice Macros — не советуем с ним связываться!) и для ваших собственных макросов, не привязанных к документам (My Macros). Учтите, что каждый документ или шаблон также считается независимым контейнером. Хранить макрос, модуль или целую библиотеку внутри документа — лучшее решение, если код



› Под редактором макросов (см. стр. 82) — окно просмотра [Watch Window], отображающее значения всех переменных во время работы макроса.



макроса должен быть видим и применяться только в этом документе (всеми, кто его откроет). Это также очень популярный способ распространения макросов, которые устанавливаются путем выбора и щелчка на них в любом другом экземпляре *LibreOffice*.

Для управления как обычными макросами и модулями, так и кодом в диалоговом окне используется то же окно Органайзера. С помощью его меню, панелей и кнопок эти действия выполняются интуитивно. Например, для перемещения или копирования кода Basic между документами и/или *LibreOffice* нужно открыть все эти файлы, вернуться в Органайзер и затем просто перетащить макросы или библиотеки из одного места в другое. Поэтому мы сосредоточимся на некоторых вещах, которые нужно знать для наиболее эффективного управления библиотеками.

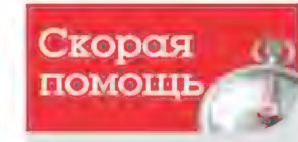
Если вы хотите создать нечто большее, чем просто набор макросов для регулярного использования, сделайте себе одолжение, и перед тем как начать, придумайте постоянную иерархию и схему именования всех макросов, модулей и т.д. Это также нужно обязательно принять в расчет, если вы хотите без проблем обмениваться своими макросами между разными пользователями и компьютерами.

Второе, что нужно знать — ваш контейнер **My Macros** располагается в подкаталоге **Basic** каталога, указанного в Tools > Options > LibreOffice > Paths [Инструменты > Настройки > LibreOffice > Пути]. Включите этот путь в свое резервное копирование, чтобы не потерять макросы в непредвиденных случаях.

Также нужно знать, что у каждого контейнера есть специальная стандартная [Standard] библиотека, которую нельзя удалить или перезаписать и которая импортируется из других контейнеров. Когда вы открываете шаблон или документ, также загружается стандартная библиотека файлового контейнера. Для активации других библиотек, которые хранятся в том же документе, т.е. чтобы включить их макросы, нужно дважды щелкнуть по их названиям в Органайзере.

Библиотеки в отдельные документы и в *LibreOffice* также можно добавить «в виде ссылки», т.е. в режиме только для чтения, поставив галочку Insert as Reference [Вставить в виде ссылки]. Разумеется, при удалении такой библиотеки вы удалите только ее локальную ссылку, а не сам код. LXF

› Вы можете велеть *LibreOffice* исполнять макросы автоматически, по наступлению каких-либо событий. Но будьте осторожны!



Всегда дважды подумайте, перед тем как писать макрос или копировать его в свои библиотеки. Часто есть лучшие и обладающие лучшей переносимостью варианты автоматизации работы в офисе.

Источники и ресурсы

Если вы продолжите изучать макросы, то обязательно узнаете приведенные здесь примеры в прекрасной работе А. Питоняка [A Pitonyak] и разработчиков документации *OpenOffice/LibreOffice*. Огромное спасибо за проделанную вами работу, мальчики и девочки!

Первые три страницы, которые вам нужно прочесть в разделе Basic сайта http://help.libreoffice.org — «Использование переменных [Using Variables]», «Выражение Dim [Dim Statement Runtime]» и «Макросы, управляемые событиями [Event-DrivenMacros]». Если вы рассчитываете употреблять макросы преимущественно в электронных таблицах, прочтите обсуждение на www.open-of-course.org/courses/mod/resource/view.php?id=826. Макросы, советы и многие другие справочные материалы доступны на www.ooomacros.org и, главное, на www.pitonyak.org/oo.php.

Shotcut: Начнем осваивать

Сергей Яремчук научит вас основам работы с видеоредактором *Shotcut*.



Наш эксперт

Сергей Яремчук
Фрилансер, автор более 1000 статей и 6 книг. 14 лет изучает Linux и пишет статьи, чтобы не забыть пройденное.



Количество видеоредакторов для Linux исчисляется десятками; правда, популярностью пользуются только некоторые. Большинство пользователей Linux устанавливают приложения из репозиториев: это очень просто и удобно, а рейтинг позволяет сориентироваться при выборе. На этом обычно и останавливаются. Но есть большое количество приложений, которых по разным причинам (в основном из-за лицензии) нет в репозиториях, но они стоят того, чтобы о них знали и с ними работали. Видеоредактор *Shotcut* (shotcut.org) как раз входит в это число.

Это очень мощное кроссплатформенное приложение с открытым исходным кодом, имеющее простой интерфейс и большое количество возможностей и не уступающее по функциям некоторым коммерческим решениям. С его помощью можно произвести все стандартные операции по обработке видео: редактировать видео (добавлять, вырезать, сохранять и т.п.), применять аудио- и видеофильтры, создавать многослойную анимацию, добавлять титры и многое другое. *Shotcut* поддерживает все типы видео/аудио файлов, обеспечиваемых библиотеками *ffmpeg* и *libav*, а также основные графические форматы. В качестве источника видео и фильтров может выступать HTML5. Из специфических функций можно отметить захват видео с web-камер, HDMI, потоков IP, экрана X11 и аудио, и трансляцию потока разными способами (HTTP, HLS, RTMP, RTSP, MMS, UDP). Для обработки OpenGL задействуется GPU, поддерживается многопоточность [multithreading]. Готовые наборы настроек, drag-n-drop и многоуровневая история правок заметно упрощают работу с приложением. Возможна автоматизация заданий и пакетное кодирование файлов.



› **Рис. 1. При первом запуске внешний вид *Shotcut* может сбивать с толку.**

Начинаем работу

В репозиториях пакет не встречается (были попытки добавить в PPA Ubuntu, но сейчас он убран), поэтому для установки необходимо использовать только то, что предлагает разработчик. Но это не сложно. На сайте проекта доступны архивы для x86/x64-битных версий Linux (Mint 12+, Ubuntu 12.04+, Debian 7+, Fedora 15+, openSUSE 12+, Arch/Manjaro). Внутри — все основные библиотеки, необходимые для работы, поэтому достаточно распаковать и щелкнуть по ярлыку ./Shotcut/Shotcut. Для удобства ярлык можно вынести на панель или в меню; в зависимости от используемого рабочего стола это делается по-разному, так что на этом останавливаться не будем. Если запуск через ярлык не сработает, можно запустить из консоли скрипт **./Shotcut/Shotcut.app/shotcut**. В этом случае сразу получим и отладочную информацию, которая может быть полезна, если что-то идет не так.

В зависимости от дистрибутива, возможно, потребуется дополнительно установить ряд библиотек и плагинов (*SDL*, *libexif*, *Jack* и *LADSPA*) — список можно найти на сайте (shotcut.org/bin/view/Shotcut/LinuxDistroPackages). Для Ubuntu/Linux Mint необходимо установить как минимум *Jack*:

```
$ sudo apt-get install jackd2
```

и плагины, которые можно найти при помощи запроса **apt-cache search ladspa**.

Интерфейс *Shotcut* написан на *Qt5*, и его можно назвать классическим для подобного рода приложений. Вдобавок он хорошо работает в мультимониторных конфигурациях и на планшетниках. Поддерживается управление при помощи Leap Motion устройства.

Понимая процесс, можно найти все, что нужно, даже при отсутствии локализации. Внешний вид меняется при помощи скинов, а количество и расположение доков можно установить как наиболее удобно. Правда, при первом запуске *Shotcut* немного сбивает с толку тех, кто ранее не работал с этой программой. Так, при загрузке будет показано лишь информационное окно Quick Start Guide, а если добавить видео — то только основной экран, напоминающий видеопроигрыватель, а не редактор. Все остальное (timeline, фильтры, плей-лист, индикатор уровня, регулятор громкости аудио и т. п.) включается/выключается в меню View или при помощи кнопок.

Также практически все операции можно активировать при помощи горячих клавиш (**shotcut.org/bin/view/Shotcut/KeyboardShortcuts**). Я бы даже сказал, что это наиболее удобный способ работы с программой, так как все кнопки сделаны слишком маленькими. Горячие клавиши позволяют очистить экран от лишних элементов и очень ускоряют процесс — правда, после того, как уже приноровишься.

При запуске будет показана небольшая подсказка по основным операциям. Многие действия интуитивны, и их можно выполнить разными способами. В итоге программу очень просто освоить и очень комфортно работать, легко подобрать для себя наиболее удобный алгоритм редактирования. Проект предлагает ряд видеоруководств и FAQ, позволяющих получить общее впечатление о работе с *Shotcut*, но для полного понимания потребуется знание английского.

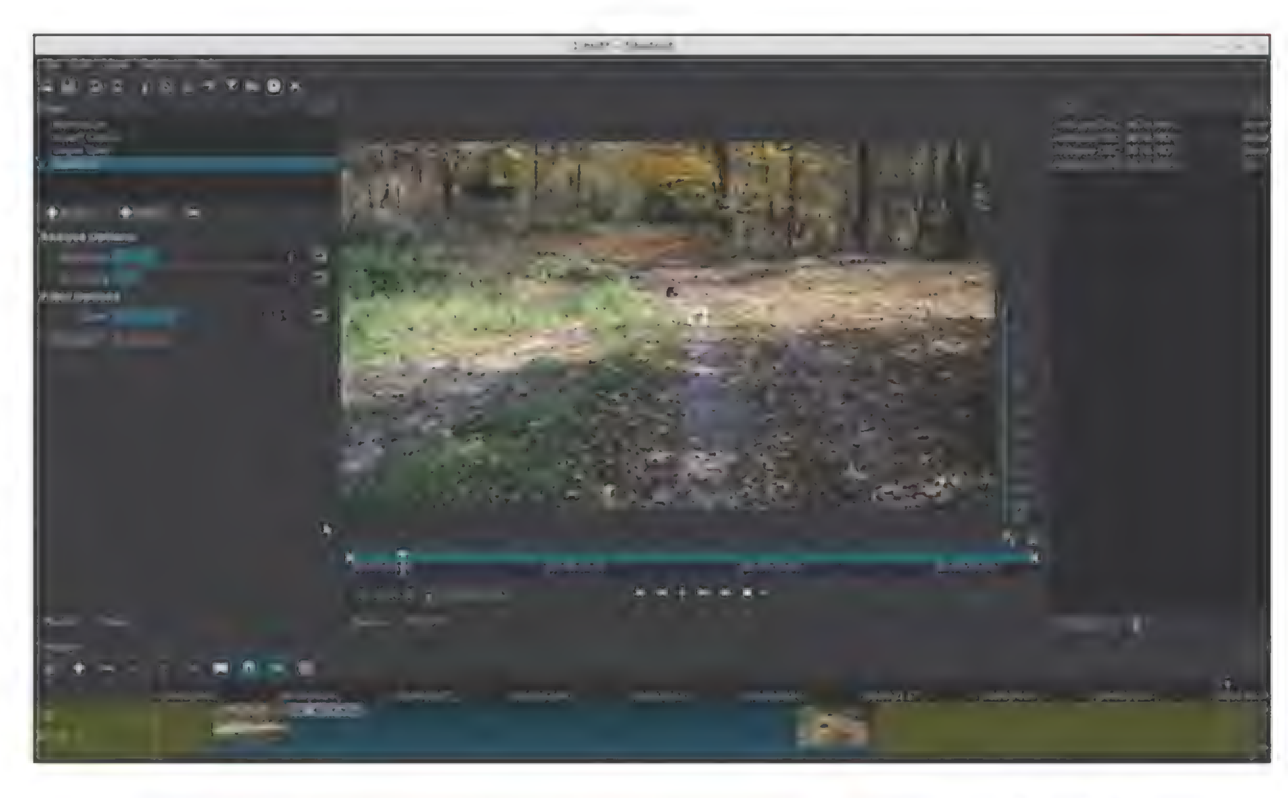
Добавляем видео в проект при помощи перетаскивания в основное окно или сразу во вкладку Playlist. Добавить также можно, нажав кнопку «+» и указав на файл; удаляются файлы из проекта при помощи клавиши Del или «-». Порядок следования видео можно изменить, просто захватив и установив в нужное место. В окне просмотра две вкладки: Source — показывает текущее видео и Program — где собраны все файлы проекта. В последнем случае начало каждого клипа помечено цифрой 1, 2, 3, ... Быстро перейти к нужному клипу в Program можно, просто щелкнув по нему дважды в окне Playlist или нажав цифру на клавиатуре, после чего курсор установится в начало.

> «Программу очень просто освоить и комфортно работать.»

Можно поступать наоборот: перетаскивать файл в окно воспроизведения программы, затем, как есть или после некоторой обработки (например, убрав лишнее), тащить видео в окно Playlist, после чего в проект добавится новый клип. Вариантов, как видим, много. Кроме видео, в проект можно добавлять изображения и аудиофайлы.

В меню File > Open Other найдется еще несколько интересных пунктов, позволяющих открыть поток, захватить видео (HDMI, Video4linux, экран), аудио (*Pulse*, *Jack* и *ALSA*) и сгенерировать видео (шум, цвет и несколько пресетов). Причем, если достаточно мощности, можно одновременно захватывать видео с экрана и работать с проектом.

Теперь можно проиграть файлы, причем можно увеличить скорость (нажав дважды L) и направление (J) воспроизведения, или при помощи мыши. Используя клавиши Page Down/Up, можно перемещаться посекундно вперед/назад.



› Рис. 2. Использование фильтра в *Shotcut*.

По умолчанию отображается весь трек, но захватив треугольники на временной шкале и сдвинув их при помощи мыши, мы можем ограничить трек определенным участком. Как вариант — запустить воспроизведение и в нужных точках нажать клавиши I (In, начало) и O (Out, конец). Перемещаться между началом и концом отмеченного участка можно при помощи клавиш со стрелками Right (или K+L) и Left (или K+J). Выделенный участок можем добавить как новый клип на Playlist, просто перетащив его туда. Или выбрать в меню файла Open As Clip, после чего произвести нужные операции, а затем выбрать в меню Update. Тогда видео в проекте обновится с произведенными установками.

Все действия в проекте отображаются в панели History; доступна отмена операций.

Нажав в верхнем меню на кнопку I (Properties), можно просмотреть и изменить некоторые свойства файла. Например, если видео имеет несколько дорожек, то во вкладке Audio в списке Track их можно поменять или совсем отключить звук (None). Здесь же ползунок Sync позволяет синхронизировать аудио со звуком. Новые установки можно сбросить, нажав Reset.

Все добавленные видео отображаются в Playlist, который можно убрать при помощи одноименной кнопки или через меню View. Также все видео, к которым недавно обращались в программе вообще, доступны в Recent, поэтому файлы не придется долго искать. Внизу каждого окна находится еще одна кнопка с изображением трех горизонтальных линий, вызывающая меню. Такая кнопка есть во многих настройках — в каждом случае пункты будут разные, поэтому просто следует с ними познакомиться. Обратите внимание, что вновь открытые окна (Recent, Playlist и т. д.) появляются в виде вкладок; но захватив их мышью, можно их отделить и переместить в любое место, разместив как удобнее. Чтобы закрыть, достаточно нажать крестик в правом верхнем углу окна.

Проект можно сохранить для дальнейшего редактирования (в файл MLT).

Также следует ознакомиться с пунктами меню Setting, в котором включается использование GPU, *Jack*, видеорежим, режим деинтерлейсинга, интерполяции, включение внешнего монитора, выбирается язык (русского, увы, пока нет) и тема. Многие операции производятся в реальном времени, поэтому наличие современного компьютера очень желательно, иначе будет подтормаживать. Но при помощи настроек в Setting можно чуть снять нагрузку с CPU: например, Interpolation установить в Nearest Neighbor. »



› Рис. 3. Настройка перехода между видео.

Использование фильтров

Добавление фильтра в проект происходит очень просто: необходимо перейти во вкладку Filter. Если она не открыта, то выбрать View > Filters. В поставке имеется 21 видеофильтр (размытие, резкость, инвертирование цвета, зеркалирование, волны, баланс белого, поворот, обрезание кадра) и 9 аудиофильтров (нормализация звука, баланс, микшер каналов), которых хватает для большинства домашних видео. Выбираем нужный фильтр в списке, после чего его название появляется в окне. Отметив мышью, мы получим доступ к дополнительным установкам. Например, при настройке баланса белого можем выбрать пресеты, указать цветовую температуру или использовать пипетку, позволяющую выбрать нужный цвет в кадре. Текущие установки сразу можем сохранить в качестве пресета для дальнейшего использования. Нормализация звука также не выглядит сложно: включаем фильтр — будут показаны текущие установки звука; затем нажимаем Analyse, далее повторяем эту операцию для всех клипов проекта. После этого звук уже не прыгает. Аналогично работает фильтр стабилизации изображения, который может пригодиться, когда производилась съемка с рук и изображение «скачет». После его активации появятся ручные настройки (Shakiness, Accuracy и Zoom) и кнопка Analyse (результат проверки будет предложено сохранить в файл). Обратите внимание, что вставка титров тоже реализована как видеофильтр — Text. После его включения появляется поле, в котором следует ввести нужный текст, выбрать шрифт и цвет и указать позицию. Последнее можно ввести вручную или воспользоваться пресетами, позволяющими установить текст в нужное место экрана без лишней возни.

› Рис. 4. Использование композитинга.



Чтобы отключить фильтр, достаточно снять флажок возле его названия.

Собственно, отметив часть видео или собрав все файлы в проект, мы можем перекодировать видео в нужный формат или запустить онлайн-трансляцию. Для этого выбираем Encode и в открывшемся окне наиболее подходящую предустановку (из более чем 30). Далее, при необходимости можем изменить все возможные установки видео и аудио (формат, кодек, разрешение, битрейт и т. д.) — просто вбиваем в поле нужное значение. Если нужно задать значение, которого нет, указываем его в Other. Поле Custom позволяет создавать свои предустановки, которые можно сохранить. Далее нажимаем Encode File для кодирования или Stream для трансляции. В последнем случае получим URL, по которому можно подключиться к потоку. По окончании обработки щелчком по файлу можем вызвать меню, из которого перейти в каталог с готовым файлом и просмотреть логи.

Пока мы рассмотрели самый простой процесс редактирования видео (удалить лишнее, использовать эффекты и титры); часто этого достаточно для подготовки домашнего видео. Но *Shotcut* является нелинейным редактором, поэтому он вполне годится и для более сложных проектов.

Использование Multitrack

Чтобы редактировать видео на нескольких треках, необходимо включить Timeline [временную шкалу]. Это можно сделать, нажав кнопку на панели или выбрав View [Просмотр] > Timeline. После этого добавляем клипы, перетаскивая их из основного окна, плей-листа программы или при помощи файлового менеджера. Если нажать кнопку «+», текущее видео из основного окна будет добавлено на Timeline. Другие кнопки позволяют добавить видео после текущего, заменить текущее видео или вставить видео в текущую позицию (видео на Timeline при этом разрывается), или удалить. Если клип кинуть на другое видео, расположенное на временной шкале, то текущее видео в этом месте будет переписано новым. Поэтому если нужно вставить кусок, но сохранить первое видео полностью, его нужно сначала разрезать и освободить достаточно места, а затем уже скомпоновать как нужно. Как и в других подобных приложениях, можем захватить за край клипа (курсор при этом изменит вид) и укоротить или растянуть его до нужных размеров.

Программа умеет автоматически генерировать переход между видео, причем сама операция выполнена интуитивно. Если два видео расположить впритык, то по окончании первого сразу начинается воспроизведение второго. Но захватив мышью стык и проведя вперед или назад, мы можем создать эффект перехода, о чем скажет появившийся перечеркнутый прямоугольник. Теперь в процессе воспроизведения одно видео плавно заменяется другим. Если оставить между видео небольшое расстояние, переход будет создан автоматически. Аналогичная ситуация — если надвинуть второе видео на первое: в месте пересечения будет автоматически сгенерирован переход. Размеры перехода также можно менять, используя мышь, и даже отделить, т. к. переход представлен как отдельный клип. Чтобы изменить настройки перехода (вид, перекрытие, аудиомикс), следует его отметить и перейти в окно Properties. Но и это не все. Если навести мышь на стыке двух видео в верхний угол конца или начала клипа, увидим точку. Ловим ее мышью и, не отпуская, тянем в сторону клипа (то есть если конец, то влево, если выбрано начало то вправо) — на треке появляется тень, а внизу показывается время. Теперь при достижении отмеченного участка

первое видео будет постепенно приглушаться, а второе, наоборот, проявляться.

В самом низу под заголовками расположен ползунок, позволяющий растянуть дорожки в ширину. Каждое видео также имеет свое меню, в котором его можно удалить, добавить как клип и разрезать (Split at play head, горячая клавиша S). Отрезанный кусок впоследствии можем обрабатывать как отдельный клип. Пустое место в дорожке также удаляется через контекстное меню, в котором единственный пункт — Remove. В общем, здесь ничего неожиданного.

Кнопка Меню позволяет добавить еще аудио- и видеодорожку, сделать вид трека больше или меньше (чтобы больше вместилось на экране), убрать/показать аудио. Новая дорожка появляется выше и имеет больший приоритет, т. е. видео на нем перекрывает нижние дорожки. Добавляются файлы на нее аналогично, только теперь можем еще перетащить клип с первой дорожки. Если добавлена аудиодорожка, то при перетаскивании на нее клипа, естественно, останется только аудиочасть: видео в проекте участвовать не будет.

Названия дорожек можем изменить, чтобы легче было ориентироваться. В самом треке видим три буквы: M (Mute) — отключает звук, H (Hide) — по сути, отключает дорожку, не удаляя ее, и C (Composite).

В релизе 14.09 улучшена функция композитинга: теперь для настройки размера, положения и прозрачности видео или графики не требуется применение фильтров. Но пока реализация еще не совсем полна: в настоящее время нет возможности регулировки размера, продолжительности и положения второй дорожки. Работает композитинг очень просто. Переносим на верхнюю дорожку материал, который будет перекрывать нижнюю, и затем включаем кнопку C. Если это будет прозрачный GIF, SVG или TGA с надписью, получим субтитр. Также можно подготовить второе видео меньшего размера в Quicktime или HTML5, которое затем поместить по верху основного.

Вывод

Как видим, в Shotcut нет ничего сложного: это удобный простой видеоредактор со всеми необходимыми функциями. LXF

Горячие клавиши Shotcut

Проиграть	L или пробел (повторное нажатие увеличивает скорость)
Пауза	K или пробел
Проиграть в обратном направлении	J
Установка границ	I и O
Переход к следующему участку	Right или K+L
Переход к предыдущему участку	Left или K+J
Вперед на секунду	Page Down
Назад на секунду	Page Up
Undo	Ctrl+Z
Redo	Ctrl+Shift+Z
Fullscreen	Ctrl+Shift+F
Playlist	
Добавить	Shift+C
Удалить	Shift+X, Del или Backspace
Вставить	Shift+V
Обновить	Shift+B
Переместить вверх	Ctrl+Up
Переместить вниз	Ctrl+Down
Выбрать клип	1 2 3 4 5 6 7 8 9 0
Timeline	
Добавить аудиотрек	Ctrl+U
Добавить видеотрек	Ctrl+Y
Укоротить трек	Ctrl+K
Сделать трек выше	Ctrl+L
Выбрать трек выше/ниже	Down/Up
Zoom In	=
Zoom Out	-
Сбросить Zoom	0
Вставить	V
Перезаписать	B
Разделить	S

ПЛИС: Откроем USB-порт

Михаил Остапкевич рассказывает, как наладить связь между компьютером и ПЛИС, а **Евгений Балдин** его слушает.



Наш эксперт

Михаил Остапкевич Романтик, очарованный компьютерами и создаваемыми в них идеальными мирами.



Наш эксперт

Евгений Балдин Физик, который действительно знает, что такое нехватка вычислительных ресурсов.

Сейчас, пожалуй, простейший способ реализовать связь между компьютером и каким-либо устройством — это воспользоваться виртуальными последовательными портами поверх интерфейса USB. В случае платы Papilio можно запрограммировать RS-232 прямо в ППВМ. Для этого необходимо:

1. Реализовать передачу байтов данных с платы на хост-машину.
2. Реализовать передачу данных с хост-машины на плату.
3. Реализовать обработку команд, поступающих с хост-машины.
4. Сгенерировать несколько синхросигналов различной частоты для передачи, приёма и обработки.

Потребуются следующие блоки:

1. **Управляющий блок** Координирует работу всех остальных блоков и обрабатывает команды, полученные с хост-машины.
2. **Блок передачи** Принимает данные как последовательность байтов от управляющего блока и побитово их передает через последовательный канал.
3. **Блок приёма** Принимает поток битов по последовательному каналу и передаёт управляющему блоку полученные данные как последовательность байтов.
4. **Блок таймера** Обеспечивает все указанные выше модули синхросигналами нужных им частот.

Блок таймера

Задача этого блока — генерация синхросигналов с различными частотами, требуемыми для блоков передачи, приёма и управления. Все они нуждаются в синхросигнале, частота которого отличается от исходной (частота кварцевого резонатора — 32 МГц). Для краткости, таймеры с заданной частотой реализуются через деление исходной частоты на некоторый коэффициент (более развитую реализацию генераторов различных частот можно найти, например, на http://www.fpga4fun.com/SerialInterface1.html), как в Листинге 1 (файл **rppu_ctrl_timer.vhd** на диске к журналу):

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
use IEEE.NUMERIC_STD.ALL;
entity rppu_ctrl_timer is
  port(
    clk_in : in std_logic; -- синхросигнал от кварца 32 МГц
    clk_out : out std_logic; -- сгенерированный сигнал нужной частоты
    counter0 : in std_logic_vector (31 downto 0)
  );
end rppu_ctrl_timer;
architecture rppu_ctrl_timer_0 of rppu_ctrl_timer is
  signal counter : std_logic_vector (31 downto 0) := "000000000000000000000000000000"; -- счётчик делителя частоты
  signal clk_inner : std_logic := '0';
begin
  process(clk_in) -- блок активируется при каждом изменении сигнала clk_in
  begin
    if rising_edge(clk_in) then -- учитываем только переходы нуля в единицу
      if(counter = 0) then -- "передний фронт" сигнала
      counter <= counter0;
      clk_inner <= '1';
      else
        counter <= counter - 1;
        clk_inner <= '0';
      end if;
    end if;
  end process;
  clk_out <= clk_inner;
end rppu_ctrl_timer_0;
```

Блок передачи

Модуль передатчика получает на вход синхросигнал, передаваемый байт данных и сигнал команды передачи байта. У модуля 2 выхода: один — для индикации события «передача байта завершена», а другой подключён к контакту ППВМ, соединённому с контактом для передачи данных по последовательному порту поверх USB. Синхросигнал берётся не исходный (32 МГц от кварца), а сформированный блоком таймера. Он соответствует частоте передачи бита информации по последовательному каналу. Плата может использовать широкий спектр частот, но для совместимости с аппаратурой хост-машины и терминальными программами будем использовать одну из стандартных (1200, 2400, 9600, 19200, 38400 или 115200). Для максимального упрощения реализации выберем одну фиксированную частоту 9600, число стоп-битов — 2 (перестраховка на всякий случай; но эта реализация будет работать и с одним стоп-битом), а всестороннюю диагностику ошибок производить не будем.

Для начала опишем логику работы модуля на более абстрактном уровне — как набор параметров и внутренних переменных, а также конечный автомат, задающий правила изменения их значений. Далее параметры будем называть сигналами, а переменные — регистрами, так как речь идёт об аппаратной реализации.

Нужны следующие сигналы и регистры:

» **Флаг готовности** Выходной однобитный сигнал блока передачи, сообщающий, что блок готов принять запрос на передачу байта по последовательному каналу. Сигнал нулевой (0), когда блок не готов, и единичный (1) — когда готов.

» **Флаг требования начать пересылку** Входной однобитный сигнал. Блок его не меняет, только считывает. Бит переводится

Передача байта по последовательному каналу

Последовательный порт RS-232, который исторически получил широкое распространение, служит для обмена данными между вычислительной машиной и разнообразными периферийными устройствами на расстояние до 15 метров. Один такой порт позволяет подключить только одно устройство.

RS-232 хорош тем, что его относительно легко запрограммировать — существенно проще, чем организовать полноценную передачу данных через USB. Поэтому, несмотря на почти полное исчезновение аппаратной реализации RS232 в современной вычислительной технике, его виртуальная реализация присутствует во многих современных устройствах, подключаемых через USB. Платой за это является низкая скорость работы.

Скорость передачи измеряется в бодах или в битах в секунду. При этом учитываются все передаваемые биты — как содержательные, так и необходимые для протокола передачи данных. Поэтому реальная скорость передачи данных бит в секунду несколько ниже скорости, измеренной в бодах.

Рассмотрим некоторую частную упрощённую схему обмена данными между вычислительной машиной и устройством: скорость передачи 9600 бод, передаётся «старт-бит», 8 бит данных и один «стоп-бит». Данные передаются устройству по одному проводу последовательно, бит за битом. Для приёма данных от устройства есть другой провод. Передача асинхронная. Это означает, что временны́е задержки между соседними передаваемыми байтами являются произвольными. Когда канал ничего не передаёт, на него выставляется уровень напряжения, соответствующий логической единице (такты 0, 1 на рисунке). Когда начинается передача байта, сначала всегда передаётся нулевой «старт-бит» (такт 2). Это позволяет приёмнику определить момент начала передачи. Далее за ним передаются 7 или 8 (в зависимости от настроек) битов передаваемого байта (такты 3–10). После них идёт 1 или 2 (в зависимости от настроек) единичных «стоп-бита» (такт 11). После этого (такт 12 и далее) можно либо перейти к передаче следующего байта (передаётся нулевой «старт-бит»), либо приостановить передачу данных на произвольный промежуток времени (уровень сигнала в канале остаётся равным логической единице).



в состояние единицы для уведомления блока пересылки, что надо начать передачу байта. Блок, задающий этот сигнал, может сбросить его обратно в состояние 0 не раньше, чем блок передачи подтвердит принятие запроса к рассмотрению (когда флаг готовности сменил состояние с 1 на 0), но, конечно, заранее, до момента, когда потребуется запросить следующую пересылку байта.

» **Байт для пересылки** Сигнал, содержащий восемь разрядов. Это также входной сигнал. Блок передачи его только читает. Его нужно задать не позже момента, когда выставляется запрос на пересылку, и потом не менять, пока блок не пересылки не сообщит, что он снова готов пересылать следующий байт.

» **Выходной последовательный канал** Однобитный сигнал, в который модуль бит за битом передаёт данные.

» **Входной сигнал синхроимпульса** Поступает из модуля таймера. Задаёт интервалы времени для передачи одного бита.

» **Регистр состояния автомата** Хранит текущее состояние конечного автомата.

» **Регистр номера** Хранит номер текущего передаваемого бита в момент передачи битов данных. При передаче старт- и стоп-битов и в момент ожидания регистр сбрасывается в нуль.

» **Флаг начала обработки запроса** Обеспечивает однократность обработки каждого поступающего запроса.

Теперь разберёмся с состояниями автомата. Для каждого из них задаются значения выходных сигналов и регистров. Здесь же перечислены условия перехода в другие состояния.

1 В состоянии ожидания блок ничего не пересылает, ждет запрос на пересылку. Когда запрос приходит, происходит переход в состояние пересылки старт-бита. Состояние ожидания является единственным, в котором флаг готовности взведён в единицу. В выходной последовательный канал выставлена единица.

2 В состоянии пересылки старт-бита в выходной канал подается нуль, регистр номера сбрасывается в нуль, и происходит переход в состояние передачи бита данных.

3 В состоянии передачи бита данных уровень сигнала выходного канала задаётся в соответствии со значением бита из байта для пересылки. Номер бита определяется регистром номера. После передачи последнего седьмого бита происходит переход в состояние передачи первого стоп-бита.

4 Состояние передачи стоп-бита меняет значение сигнала выходного канала на единицу и переходит в состояние передачи второго стоп-бита. Из состояния передачи второго стоп-бита происходит переход в состояние ожидания.

Предложенный вариант автомата далеко не единственный. В более полной реализации следует предусмотреть состояния ошибки и восстановления после неё.

На основании этих вводных получим такой исходный текст на VHDL (файл **rppu_io_serial_sender.vhd** на диске):

```
<4 заголовочные строки как в Листинге 1>
entity rppu_io_serial_sender is
  port(
    hw_tx : out std_logic; -- внешняя линия передачи данных
на хост-машину
    rate_clk : in std_logic; -- входной синхросигнал, 9600 Гц
    data : in std_logic_vector (7 downto 0); -- байт для передачи
    request_start : in std_logic; -- сигнал требования начать
передачу
    is_ready : out std_logic -- сигнал завершения передачи
(байт передан)
  );
end rppu_io_serial_sender;
architecture Behavioral of rppu_io_serial_sender is
  type states is ( -- тип, перечисляющий состояния конечно-
го автомата
    ready, sending_startbit, sending_databit, sending_stopbit1,
sending_stopbit2);
  signal state : states := ready; -- регистр текущего состояния
автомата
    signal request_start_handled: std_logic := '0'; -- флаг начала
обработки запроса
  signal num_bit: std_logic_vector (2 downto 0); -- рег. номера
передаваемого бита
begin
  process (rate_clk) begin
    if(rising_edge(rate_clk)) then
      if (request_start = '0') and (request_start_handled = '1') then
        request_start_handled <= '0'; -- сброс флага после
сброса сигнала запроса
      end if;
      case state is
        when ready => -- состояние ожидания, в нем ничего
не передается
          hw_tx <= '1'; num_bit <= "000"; is_ready <= '1';
          if (request_start = '1') and (request_start_handled = '0')
then
```

»

```
            request_start_handled <= '1'; -- при поступлении
запроса переходим
            state <= sending_startbit; -- к передаче старт-бита
          else
            state <= ready; -- иначе остаёмся в состоянии
ожидания
          end if;
        when sending_startbit => -- состояние передачи
старт-бита
          hw_tx <= '0'; state <= sending_databit; -- далее будем
передавать данные
          is_ready <= '0'; num_bit <= «000»; -- начнём с нулевого
бита
        when sending_databit => -- состояние посылки бита
данных
          hw_tx <= data(conv_integer(unsigned(num_bit))); --
шлём бит с номером,
          is_ready <= '0'; -- взятым из регистра
          if num_bit = «111» then
            state <= sending_stopbit1; -- после посылки
последнего бита перейдём
            num_bit <= «000»; -- к посылке стоп-бита
          else
            state <= sending_databit;
            num_bit <= num_bit + 1; -- далее будем слать
следующий бит
          end if;
        when sending_stopbit1 => -- состояние посылки первого
стоп-бита
          hw_tx <= '1'; state <= sending_stopbit2; -- после
посылки
          num_bit <= "000"; is_ready <= '0';
        when sending_stopbit2 => -- состояние посылки второго
стоп-бита
          hw_tx <= '1'; state <= ready;     -- далее заканчиваем
пересылку,
          num_bit <= «000»; is_ready <= '0'; -- переходим
в состояние ожидания
      end case;
    end if;
  end process;
end Behavioral;
```

Блок приёма

В этом блоке набор сигналов, регистров, а также управляющий автомат в основном соответствуют определениям, введённым для блока передачи. Отметим только различия:

» Условием для запуска приема байта является не управляющий сигнал, а понижение сигнала во входном последовательном канале, которое означает старт-бит.

» Для своевременного обнаружения старт-бита необходимо опрашивать состояние входного канала существенно чаще (в 8–16 раз), чем скорость передачи канала (oversampling).

» Управляющий автомат имеет меньше состояний. В частности, нет состояния считывания старт-бита, так как это происходит при выходе из состояния ожидания.

Реализация (файл **rppu_io_serial_receiver.vhd** на диске):

```
<4 заголовочные строки как в Листинге 1>
entity rppu_io_serial_receiver is
  port(
    hw_rx : in std_logic; -- аппаратный входной сигнал от по-
следовательного порта
    rate_clk : in std_logic; -- синхроимпульс с частотой в 8 раз
выше 9600
    data : out std_logic_vector (7 downto 0); -- считанный байт
    is_ready : out std_logic -- данные получены, можно
считывать
  );
end rppu_io_serial_receiver;
architecture Behavioral of rppu_io_serial_receiver is
  type states is ( -- состояния автомата
    waiting, -- состояние ожидания стоп-бита
    receiving_databit, -- состояние приёма бита данных
    receiving_stopbit -- состояние пропуска стоп-бита перед пе-
реходом в ожидание
  );
  signal state : states := waiting; -- текущее состояние автомата
  signal num_bit : std_logic_vector (2 downto 0); -- номер считы-
ваемого бита данных
  signal baud_counter : std_logic_vector (2 downto 0) := "000";
  signal baud_tick : std_logic := '0'; -- индикация истечения ин-
тервала 1/9600 сек.
begin
  process(rate_clk) begin
    if(rising_edge(rate_clk)) then
      if (hw_rx = '0') and (state = waiting) then
        is_ready <= '0'; baud_counter <= "111";
        state <= receiving_databit; baud_tick <= '0';
      else
        baud_counter <= baud_counter - 1;
        if baud_counter = 0 then
          baud_tick <= '1';
        else
          baud_tick <= '0';
        end if;
      end if;
      if baud_tick = '1' then
        case state is
          when waiting =>
            num_bit <= "000"; is_ready <= '1';
          when receiving_databit =>
            data(conv_integer(unsigned(num_bit))) <= hw_rx;
            is_ready <= '0';
            if num_bit = "111" then
              state <= receiving_stopbit; num_bit <= "000";
            else
              state <= receiving_databit; num_bit <= num_bit + 1;
            end if;
          when receiving_stopbit =>
            is_ready <= '0';
            num_bit <= "000"; state <= waiting;
        end case;
      end if;
    end if;
  end process;
end Behavioral;
```

Управляющий блок

Управляющий блок координирует работу всех остальных блоков, связывая их входы и выходы и образуя из них единую систему, а также реализует простейший протокол взаимодействия хост-машины и платы, в рамках которого плата принимает символы от хост-машины. Два символа ('1', '2') соответствуют командам «включить светодиод» и «выключить светодиод». Плата в ответ выполняет нужное действие и возвращает хост-машине символ '+'. В ответ на другие входящие символы плата возвращает '-'.

Реализация протокола (файл **rppu_ctrl_unit.vhd** на диске):

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity rppu_ctrl_unit is
```

```
  port(
    clk_fastest : in std_logic;
    clk_slowest : in std_logic;
    led0 : out std_logic;
    serial_sender_launch : out std_logic;
    serial_sender_ready : in std_logic;
    serial_receiver_ready : in std_logic;
    data_to_send : out std_logic_vector (7 downto 0);
    data_received : in std_logic_vector (7 downto 0)
  );
end rppu_ctrl_unit;
architecture Behavioral of rppu_ctrl_unit is
  type states is (ready, start_req, error);
  signal state : states := ready;
  signal reg_data : std_logic_vector (7 downto 0) := "00100101";
  signal blink : std_logic := '0'; -- флаг режима мигания
  signal blinker : std_logic := '0'; -- фаза мигания
  signal handled : std_logic := '0'; -- флаг однократной
обработки входного символа
begin
  process(clk_slowest) begin -- смена фаз мигания
    if rising_edge(clk_slowest) then
      blinker <= not blinker;
    end if;
  end process;
  process(clk_fastest) begin
    if rising_edge(clk_fastest) then
      if serial_receiver_ready = '1' and handled = '0' then
        handled <= '1'; -- обеспечивает однократность
обработки символа
        case data_received is
          when «00110001» => -- поступил символ единицы
            reg_data <= «00101011»; -- послать обратно «+»
            led0 <= '1'; -- зажигаем светодиод
            blink <= '0';
          when «00110010» => -- поступил символ двойки
            reg_data <= «00101011»; -- послать обратно «+»
            led0 <= '0'; -- гасим светодиод
            blink <= '0';
          when «00110011» => -- поступил символ тройки
            reg_data <= «00101011»; -- посылаем обратно «+»
            blink <= '1'; -- включаем режим мигания
            led0 <= blinker;
          when others => -- прибыл символ неизвестной команды
            reg_data <= "00101101"; -- послать обратно"–"
        end case;
        if state = ready then
          if serial_sender_ready = '1' then
            state <= start_req;
          else
            state <= error;
          end if;
        end if;
      else
        if serial_receiver_ready = '0' and handled = '1' then
          handled <= '0'; -- начинается приём нового символа,
нужно сбросить флаг
        end if;
        if blink = '1' then -- обработка режима мигания
          led0 <= blinker;
        end if;
      end if;
      case state is
        when ready =>
          serial_sender_launch <= '0';
        when start_req =>
          serial_sender_launch <= '1';
          -- ждем подтверждения принятия запроса к обработке
          if serial_sender_ready = '1'then
            state <= start_req;
          else
            state <= ready;
          end if;
        when error =>
          serial_sender_launch <= '0';
          -- ожидание готовности передатчика
          if serial_sender_ready = '1' then
            state <= ready;
          else
            state <= error;
          end if;
      end case;
    end if;
  end process;
  data_to_send <= reg_data;
end Behavioral;
```

Результирующая программа, включающая все блоки, вынесена на диск (файл **rppu_assembly.vhd**).

Сборка всех блоков

Создаём проект аналогично прошлому уроку (**LXF188**, стр. 80). Добавляем в него все приведённые выше листинги. Немного изменим файл ограничений — ведь у нас теперь два новых внешних сигнала, приём (rx) и передача (tx):

```
NET "clk" LOC = "P89" | IOSTANDARD=LVCMOS25 | PERIOD =
31.25ns;
NET "p17" LOC = "P17" | IOSTANDARD=LVTTL;
NET "tx" LOC="P90" | IOSTANDARD=LVTTL | DRIVE=8 |
SLEW=FAST;
NET "rx" LOC="P88" | IOSTANDARD=LVTTL | DRIVE=8 |
SLEW=FAST | PULLUP;
```

Для светодиода можно выбрать и другой контакт, но для сигналов tx, rx контакты изменить нельзя: именно они соединены с выводами другой микросхемы на плате, служащей интерфейсом между микросхемой ППВМ и последовательным портом.

Хост-программа

На стороне хоста в простейшем случае можно использовать программу терминала, например, *miniterm*. Запускаем ее с ключом **-s**, выбираем имя порта (обычно это **/dev/USB0** или **/dev/USB1**) и задаем его параметры: скорость 9600 бод, 8 передаваемых бит в байте, без бита четности (No Parity), 1 стоп-бит (9600 8N1).

```
> minicom -s
+-----------------------------------------------------------+
| A - Последовательный порт        : /dev/ttyUSB1           |
| B - Размещение lock-файла        : /var/lock              |
| C - Программа при выходе         :                        |
| D - Программа при запуске        :                        |
| E - Скорость/Чётность/Биты       : 9600 8N1               |
| F - Аппаратное управление потоком  : Да                   |
| G - Программное управление потоком : Нет                  |
|                                                           |
|   Какую настройку изменить?                               |
+-----------------------------------------------------------+
```

Послесловие

Представленная реализация минимальна (без обработки и индикации ошибок, настройки параметров последовательного канала и интерфейса для передачи пакетов данных), но она работает. Все исходные тексты см. также на http://qwertus.com/fpga/p2.html. LXF

Minecraft: Импорт изображений

Вы когда-нибудь хотели трансформировать изображения в квадратные блоки с палитрой из 16 цветов? Нет? Это непросто, но **Джонни Бидвелл** справится!



Наш эксперт

Джонни Бидвелл планирует включить в *Minecraft* игру *2048*. Это может отрицательно сказаться на производительности.





› **Это не какое-то кровавое облако, а гигатская ягодка-малинка, плавающая в небе. Обычный рабочий день в офисе.**

Технологии избаловали нас 32-битными многомегапиксельными изображениями. Помните эти блочные спрайты былых времен, когда приходилось призывать на помощь воображение, чтобы воссоздать картинку по этим гигантским пикселям? На нашем уроке мы вернемся в безмятежные времена мира *Minecraft* и покажем, как импортировать и отображать изображения в виде блоков из цветной шерсти. И добавим Python. Да еще Raspberry Pi.

Самые красочные блоки в *Minecraft* — шерстяные (тип блока 35): доступно 16 различных цветов, назначаемых параметром **blockData**. На нашем уроке мы будем пользоваться ими, но вы можете немного доработать программу и использовать другие блоки, добавив в свою палитру новые цвета. Процесс уменьшения палитры изображения — это пример квантизации: из изображения удаляется информация, и оно занимает меньше байт. Чтобы выполнить это квантование цвета, сначала нужно определить новую ограниченную палитру, задав красный, зеленый и синий компоненты для каждого из 16 цветов шерсти. Это скучный процесс, включающий импорт изображения каждого цвета шерсти в *GIMP* и определение средних значений компонентов с помощью «пипетки», но к счастью, кое-кто уже сделал за нас эту работу.

Стандартная установка

Если вы следите за нашей серией о *Minecraft: Pi Edition*, процедура установки должна быть вам знакома, но если нет, ниже вы прочтете о том, как установить *Minecraft* и скопировать API для использования в своем коде.

Мы предполагаем, что у вас Raspbian и система получила все обновления. Загрузите *Minecraft* с http://pi.minecraft.net, затем откройте терминал и распакуйте файл следующим образом (предполагается, что вы загрузили файл в свой домашний каталог):

```
$ tar -xvzf ~/minecraft-pi-0.1.1.tar.gz
```

Все файлы окажутся в подкаталоге под названием **mcpi**. Чтобы запустить *Minecraft*, сначала запустите X, а затем в терминале выполните команды:

```
$ cd ~/mcpi
$ ./minecraft-pi
```

Для проекта на *Minecraft* удобно настроить рабочий каталог и скопировать API туда. Архив на диске распакуется в каталог **mcimg**, поэтому можете распаковать его в свой домашний каталог и скопировать файлы API командами

```
$ tar -xvzf mcimg.tar.gz
$ cp -r ~/mcpi/api/python/mcpi ~/mcimg/minecraft
```

На нашем уроке мы воспользуемся PIL (Python Imaging Library — библиотека для работы с изображениями Python), которая уже устарела, но ее более чем достаточно для скромных требований данного проекта. Она умеет импортировать изображения в форматах **.jpg**, **.png** и других, поэтому возиться с преобразованием изображений не придется. Установите ее командой

```
$ sudo apt-get install python-imaging
```



› Имея всего 16 цветов, Стив может нарисовать все, что захочет (но не слишком детально).

Также нужно изменить размер изображения — в мире *Minecraft* по каждому измерению всего 256 блоков, и поскольку каждый пиксель будет преобразован в один блок, наше изображение должно быть не более 256 пикселей по длинной стороне. Однако вы можете не захотеть, чтобы изображение занимало все это место, да и блоки нельзя громоздить друг на друга более 64 штук в высоту, поэтому приведенный код изменит размер вашего изображения до 64 пикселей по длинной стороне, сохранив исходное соотношение сторон. Это поведение можно изменить с помощью переменной **maxsize**, но если изображение слишком высокое, его верхушка будет обрезана.

Модуль PIL выполняет квантование и изменение размера всего одной строкой, но сначала надо задать палитру и вычислить новый размер изображения. Палитра представляется списком значений RGB, которые затем дополняются нулевыми значениями до 256 цветов. Для удобства мы будем указывать цвета в порядке, соответствующем параметру blockData.

```
mcPalette = [
  221,221,221, # White [Белый]
  219,125,62, # Orange [Оранжевый]
  179,80,188, # Magenta [Малиновый]
  107,138,201, # Light Blue [Голубой]
  177,166,39, # Yellow [Желтый]
  65,174,56, # Lime Green [лимонно-зеленый]
  208,132,153, # Pink [Розовый]
  64,64,64, # Dark Grey [Темно-серый]
  154,161,161, # Light Grey [Светло-серый]
  46,110,137, # Cyan [Бирюзовый]
  126,61,181, # Purple [Лиловый]
  46,56,141, # Blue [Синий]
  79,50,31, # Brown [Коричневый]
  53,70,27, # Green [Зеленый]
  150,52,48, # Red [Красный]
  25,22,22, # Black [Черный]
]
mcPalette.extend((0,0,0) * 256 - len(mcPalette) / 3)
```

К сожалению, на коде с диска отсутствует "/ 3" — это всего лишь опечатка, не имеющая настоящих последствий (фи...). У такого дополнения палитры есть один потенциально нежелательный недостаток: полное исчезновение черных пикселей из вашего изображения. Это потому, что их значение ближе к абсолютно черному цвету (которым мы искусственно расширяем палитру), чем чуть более светлый цвет «черной» шерсти. Чтобы это обойти, можете изменить (0,0,0) выше на (25,22,22), чтобы абсолютно черные цвета исчезли. Если вы работаете с прозрачным изображением, стоит заменить это значение фоновым цветом вашего изображения, чтобы прозрачные части изображения не были заполнены. Для хранения этой палитры создадим новое изображение размером в один пиксель:

```
mcImagePal = Image.new("P", (1,1))
mcImagePal.putpalette(mcPalette)
```

В прилагаемом архиве есть файл **test.png** — это вообще-то талисман Scratch, но вы вполне можете заменить его своим изображением, чтобы проверить, как оно пройдет изменение размера и квантование. А если не повезет, продукт всегда можно взорвать к чертям. Чтобы сохранить соотношение сторон, мы используем при делении функцию **float**, во избежание округления до целого числа.

```
mcImage = Image.open("test.png")
width = mcImage.size[0]
height = mcImage.size[1]
ratio = height / float(width)
maxsize = 64
```

Как мы упоминали ранее, блоки в *Minecraft* нельзя размещать по вертикали более 64 подряд (видимо, по технике безопасности). Следующий блок кода пропорционально изменяет размер изображения до 64 пикселей по длинной стороне.

```
if width > height:
  rwidth = maxsize
  rheight = int(rwidth * ratio)
else:
  rheight = maxsize
```

»

```
rwidth = int(rheight / ratio)
```

Если у вашего изображения длина значительно превышает высоту, можете использовать более 64 пикселей в горизонтальном направлении и зафиксировать высоту на 64. Заменив блок выше всего двумя строками с **else**, можно добиться именно этого результата.

Теперь преобразуем изображение в цветовое пространство RGB, чтобы не путать функцию **quantize()** с информацией о прозрачности, и затем изменим его в соответствии с новой палитрой и новыми размерами. Вы можете получить лучшие результаты, сначала изменив размер и затем выполнив квантование, но мы предпочитаем делать все по алфавиту.

```
mcImage = mcImage.convert("RGB")
mcImage = mcImage.quantize(palette = mcImagePal)
mcImage = mcImage.resize((rwidth,rheight))
```

Для простоты поместим изображение рядом со Стивом, точнее, в пяти блоках от него, и выровняем его по направлению X. Если Стив находится на краю мира на положительном конце оси X или высоко на холме, некоторые части изображения, к сожалению, будут утеряны. Получить координаты Стива просто:

```
playerPos = mc.player.getPos()
x = playerPos.x
y = playerPos.y
z = playerPos.z
```

Затем достаточно пройти в цикле по обеим измерениям нового изображения, употребив медленный, но верный метод **getpixel()** для получения индекса в нашей палитре и рисуя должный цвет в должном месте с помощью функции **setBlocks()**.

Если у вашего изображения имеется канал прозрачности, то функция **getpixel()** вернет для прозрачных пикселей None, и блоков нарисовано не будет. Чтобы изменить это поведение, можно добавить выражение **else** и рисовать блок фонового цвета по умолчанию. Координаты изображения начинаются с (0,0) в левом верхнем углу, и чтобы не нарисовать изображение вверх ногами, мы будем вычитать переменную цикла **k** из **rheight**.

```
for j in range(rwidth):
  for k in range(rheight):
    pixel = mcImage.getpixel((j,k))
    if pixel < 16:
      mc.setBlock(j + x + 5, rheight - k + y, z, 35, pixel)
```

Чтобы волшебство сработало, запустите *Minecraft* и переместите Стива в положение, соответствующее вашему изображению. Затем откройте терминал и выполните команды

```
$ cd ~/mcimg
$ python mcimg.py
```

Эти объяснения перекрывают код на диске, но вы можете изрядно повеселиться, развив данную идею. Для начала стоит выделить содержимое **mcimg.py** в функцию. Эта функция может иметь несколько аргументов. Приведенный ниже вариант удобен тем, что позволяет указать файл изображения и желаемые координаты:

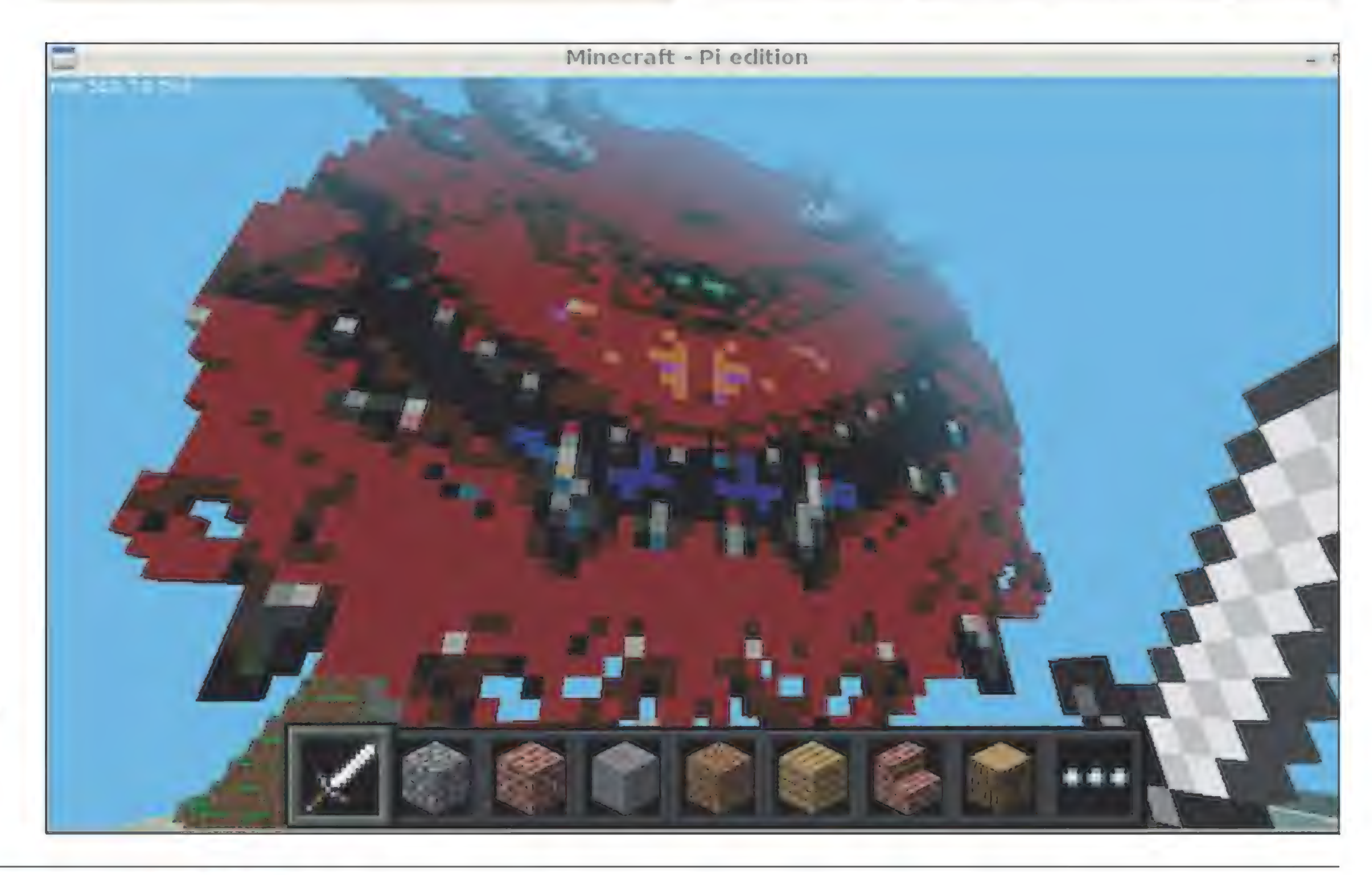
```
def drawImage(imgfile, x=None, y=None, z=None):
  if x == None:
    playerPos = mc.player.getPos()
    x = playerPos.x
    y = playerPos.y
    z = playerPos.z
```

Если координаты не указаны, используется положение игрока. Если у вас имеются разрушительные наклонности, для красных пикселей изображения можете применить взрывчатку TNT. Просто замените вызов **mc.setBlock** в цикле рисования на следующий:

```
if pixel == 14:
  mc.setBlock(j + x + 5, rheight - k + y, z, 46, 1)
else:
  mc.setBlock(j + x + 5, rheight - k + y, z, mcPaletteBlocks[pixel])
```

Если вам не понравится получившаяся картинка — на ваше счастье, она очень уязвима, и несколько точных щелчков на блоках со взрывчаткой проделают в нем несколько отверстий или превратят его в пыль. Степень разрушений зависит от красноты исходного изображения.

В полном *Minecraft* огромный набор цветных блоков, включая пять различных типов деревянных планок и лестниц, шесть видов



› В отличие от *Doom*, эта клубничка/какодемон не испускает языки пламени. Ну и хорошо.

» **Было в номере LXF186** Как сделать в Minecraft пушку на Python.



› Да, Стив, начни с лодыжек. Без них далеко не убежишь!

камня, изумруд и 16 разновидностей цветного стекла, а редакция для Pi несколько ограничена. Но есть несколько хороших кандидатов, которые расширят вашу палитру:

Название блока	ID блока	Красный	Зеленый	Синий
Золото	41	241	234	81
Ляпис-лазурь	22	36	61	126
Песчаник	24	209	201	152
Лед	79	118	165	244
Алмаз	57	116	217	212

До сих пор мы могли легко получить цвет, поскольку индекс **mcPalette** был аккуратно выравнен с параметром blockData цветной шерсти. Раз мы добавляем другие типы блоков, все усложняется, и потребуется таблица преобразования. Просто добавив новые цвета в конец существующей палитры **mcPalette** —

```
mcPalette = [ ...
  241,234,81,
  36,61,126,
  209,201,152,
  118,165,244,
  116,217,212
]
mcPaletteLength = len(mcPalette / 3)
```

мы можем сделать таблицу преобразования такой:

```
mcLookup = []
for j in range(16):
  mcLookup.append((35,j))
mcLookup += [(41,0),(22,0),(24,0),(79,0),(57,0)]
```

Таким образом, в списке **mcLookup** содержатся тип блока и данные блока для каждого цвета нашей палитры. Теперь у нас на феноменальные 31,25% больше цветов [полная гамма, — *ред.*], с которыми можно позабавиться. Чтобы воспользоваться ими в цикле рисования, используйте следующий код в циклах **for**:

```
pixel = mcImage.getpixel((j,k))
if pixel < mcPaletteLength:
  bType = mcLookup[pixel][0]
  bData = mcLookup[pixel][1]
  mc.setBlock(j + x + 5, rheight - k + y, z, bType, bData)
```

Так можно добавить в палитру любые необходимые блоки, но будьте осторожны с лавой и водой: их приятные насыщенные оранжевый и синий имеют печальную тенденцию превращаться в лаву/водопады. Попутно, соединившиеся лава и вода образуют обсидиан. Холодный твердый обсидиан. LXF

Больше измерений

Один из самых старых документов про вывод произвольных изображений в *Minecraft:Pi Edition* — отличное руководство Дэва Стотта [Dav Stott] об отображении карт Государственной географической службы, http://bit.ly/1lP20E5. Двумерные изображения — это прекрасно, но у Стива есть масса других осей...

С этой целью вышеупомянутая Служба предоставила (для полной версии *Minecraft*) мир, содержащий большую часть Великобритании с масштабом 50 м на блок. Ее датские коллеги выпустили похожую карту, хотя некоторые части мира *Minecraft*-Дания были злостно недоделаны.

Еще один отличный пример — впечатляющий проект 3D-моделирования Мартина О'Хэнлона [Martin O'Hanlon]. Он может импортировать файлы **.obj** (текстовые файлы с данными о вершинах, гранях и текстурах) и отображать их в Minecraft: Pi Edition. Читайте об этом проекте на http://bit.ly/1sutoOS.

Конечно, существует и временно́е измерение, и вы можете расширить наше руководство на него, дав Стиву несколько анимированных gif-картинок, через которые он будет перепрыгивать. Если вы захотите сделать это, все придется сделать довольно маленьким — процесс отрисовки довольно медленный и мучительный. Как и следовало ожидать, кто-то (а именно Генри Гарден [Henry Garden]) уже взялся за это и написал *Redstone* — интерфейс Clojure для *Minecraft*, который позволяет рендерить видеоролики.

Полную презентацию, включая представленную в виде блоков заставку из *Симпсонов*, см. на http://bit.ly/1sO0A2q.

Hack: Hip Hop Virtual Machine

Ярый хулитель PHP **Джонни Бидвелл** променял послеобеденный сон на изучение зрелого языка.



Наш эксперт

Джонни Бидвелл утверждает, что всю его рассеянность так или иначе можно списать на хип-хоп.



Если отвлечься от сомнительного отношения Facebook к приватности данных и от того, что это площадка назойливой саморекламы, Facebook достоин похвалы за его усилия по части PHP. Недавно на конференции OSCON в Портленде (куда мы отправили нашего Мэтта Хэнсона, чтобы он смешался с толпой, насладился бесплатным пивом и посозерцал места из телесериала «Гримм») Facebook анонсировал спецификацию формата PHP 5.6. Хотя эта спецификация мгновенно ничего не изменит, она по крайней мере документирует особенности языка и формирует почву для новых (и, надеемся, лучших) реализаций языка. Кроме того, Facebook представил HipHop Virtual Machine (HHVM), JIT-компилятор для PHP, с существенным приростом производительности по сравнению с обычным, громоздким интерпретируемым языком. Довольно удобно, если вы хотите всучить свою рекламу миллиарду человек.

Код HHVM открыт, поэтому он пригодится и более широкому сообществу PHP (хотя обычно ему нужна другая помощь). На момент написания статьи HHVM более или менее совместим с большинством фреймворков PHP, включая *WordPress*, *Drupal*, *Joomla* и *MediaWiki*. Результаты тестирования на совместимость можно узнать на http://hhvm.com/frameworks. Если тест не набрал 100 %, это не значит, что HHVM откажется работать — модульные тесты покрывают даже такие мелочи, как форматирование сообщений об ошибках.

В апреле Facebook анонсировал Hack, язык программирования для HHVM, который легко интегрируется в PHP, но предоставляет дополнительные функции, и благодаря объявлению типов и другим возможностям зрелых языков позволяет придать программистам-раздолбаям чуть больше дисциплины. Установите все, что нужно, воспользовавшись инструкциями (см. «Установка», стр. 97). Для проверки установленной HHVM сохраните следующий код в файле hello.hack:

```
<?hh
echo "Hello world, from HHVM version " . HHVM_VERSION . «\n»;
```

И запустите его командой **hhvm hello.hack**. Если все хорошо, вы увидите дружелюбное приветствие. Помните, что в HHVM можно запускать и (почти) любой код на PHP, который с добавлением константы HHVM_VERSION и представляет собой приведенный выше листинг. Обратите внимание на отсутствие закрывающего тэга ?>.

Быстрая факторизация

В нашей увлекательной статье о криптографии (см. стр. 56) мы обсуждали относительную сложность разложения на множители (факторизации) больших целых чисел. Вот вам код посложнее, который решает эту проблему:

```
<?php
function primefactor($num) {
  $sqrt = sqrt($num);
  for ($i = 2; $i <= $sqrt; $i++) {
```



› Логотипы HHVM и Hack демонстрируют явное отсутствие воображения у авторов.

```
    if ($num % $i == 0) {
      return array_merge(primefactor($num/$i), array($i));
    }
  }
  return array($num);
}
function primeloop() {
  print_r(primefactor(pow(2,61)+1));
}
primeloop();
?>
```

Этот код проверяет на простоту число $2^{61}+1$, деля его на все числа подряд до его квадратного корня. Оказалось, что это число (как и попытка приблизиться к максимально возможному 64-битному целому числу) — не простое, как видно из результата запуска скрипта:

```
$ php -f prime.php
Array
(
  [0] => 768614336404564651
  [1] => 3
)
```

Не считая самого себя и единицы, это число делится на 3 и на большое простое число.

Более быстрый PHP! Убойно!

Проницательные читатели поинтересуются, в чем смысл отдельной функции primeloop. Причина в том, что HHVM не компилирует в JIT главные функции (так называемые «псевдоглавные функции»), поэтому мы не увидим преимуществ после преобразования этого кода в Hack. Аккуратно обернутый в функцию, наш вызов **primefactor()** еще раз попадет в HHVM и, надеемся, будет выполняться быстрее. Конечно, код можно было оптимизировать и более традиционными способами: не проверяя четные множители, кроме двойки, можно было бы ускорить код вдвое; да есть и другие, менее дубовые алгоритмы проверки простоты числа. Но идея не в этом: в текущем состоянии на обветшалом компьютере автора этой статьи команда **time** выполняется за прискорбные 90 секунд. Однако стоит запустить

```
$ time hhvm prime.php
```

и мы получим более впечатляющие 30 секунд. Компилятор JIT увеличил производительность втрое.

У Hack есть два варианта: Strict и Partial. Как можно предположить, для соответствия режиму Strict [строгий] требуется больше усилий по сравнению с Partial [частичный]. На самом деле, любой PHP-код также считается Partial-разновидностью Hack. Для включения режима Strict нужно изменить начальный тэг таким образом:

```
<?hh // strict
```

PHP Version 5.5.9-1ubuntu4.3



System	Linux vbvntv 3.13.0-24-generic #47-Ubuntu SMP Fri May 2 23:30:00 UTC 2014 x86_64
Build Date	Jul 7 2014 16:33:02
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-gd.ini, /etc/php5/apache2/conf.d/20-imagick.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/20-readline.ini
PHP API	20121113
PHP Extension	20121212
Zend Extension	220121212
Zend Extension Build	API220121212,NTS
PHP Extension Build	API20121212,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.5.0, Copyright (c) 1998-2014 Zend Technologies
with Zend OPcache v7.0.3, Copyright (c) 1999-2014, by Zend Technologies



› Функция phpinfo() в HHVM выводит только слово 'HipHop' — не так, как в старом добром PHP.

По умолчанию используется режим Partial; это можно указать явно, добавив **//partial**, но делать так не рекомендуются. В Hack есть утилита командной строки *hackificator*, которая делает часть грязной работы по преобразованию с PHP на Hack. Запустить ее на нашем тривиальном примере (с двойной порцией ругательств по адресу языка) можно командой:

```
$ hackificator -hackify prime.php
```

Нам сообщат, что требуется заменить начальный тэг <?php на <?hh. Но мы это уже знаем. Пакет Hack также содержит »

Установка

В целом, у Hack и HHVM есть несколько зависимостей, которые все время меняются, поэтому они вряд ли попадут в стандартные репозитории вашего дистрибутива. Но не бойтесь — на официальных зеркалах есть пакеты для Debian, Fedora, Mint и Ubuntu. Для Arch Linux в AUR есть пакет *hhvm*, а для Gentoo необходимые сборки можно найти в *javer*.

В Ubuntu нужно добавить следующую строку в файл **/etc/apt/sources.list** (или добавить файл с этим содержимым в **/etc/apt/sources.d**):

```
deb http://dl.hhvm.com/ubuntu trusty main
```

Затем добавьте открытый ключ HHVM в ключевую пару Apt:

```
wget -O - http://dl.hhvm.com/conf/hhvm.gpg.key |
sudo apt-key add -
```

Теперь можете устанавливать и обновлять пакеты:

```
apt-get update
apt-get install hhvm
```

В Debian и Mint используются небольшие вариации этих команд, но в Fedora (19 и 20) нужно создать файл **/etc/yum.repos.d/hhvm.repo** со следующим содержимым:

```
[hhvm]
name=HHVM for Fedora $releasever - $basearch
baseurl=http://dl.hhvm.com/
fedora/$releasever/$basearch/
enabled=1
gpgcheck=0
```

Это позволит вам установить все необходимое просто командой

```
sudo yum install hhvm
```

Framework	Latest	Best Recent	History
assetic	100	100	
cakephp3	96.59	96.59	
codeigniter	100	100	
composer	100	100	
doctrine2	100	100	
drupal	99.98	100	
ezpublish-kernel	99.78	99.78	
facebookphpsdk	100	100	
fastroute	100	100	
guzzle	100	100	
guzzle-streams	98.61	98.61	
guzzle4	92.54	92.54	
hhvmquicktests	100	100	
html5php	100	100	
idiorm	100	100	

docs.hhvm.com

› **HHVM старается угодить всем, кто пользуется PHP, а не только Facebook, поэтому он поддерживает многие популярные фреймворки.**

утилиты *hh_client* и *hh_server*, которые совместно могут следить за каталогом проекта в реальном времени и предупреждать вас каждый раз, когда вы нарушили правила. Чтобы воспользоваться ими, поместите пустой файл **.hhconfig** в корневой каталог своего проекта.

Hack не слишком любит массивы, предпочитая более специализированные варианты коллекций: Map [карты], Vector [векторы], Set [наборы] или Pair [пары]. Для нашего примера больше всего подходит конструкция Vector. Поняв, как устроены векторы, можно начать набирать переменные. Hack позволяет (а в режиме Strict заставляет) объявлять типы членов класса, параметры функций и возвращаемые значения. Статическая типизация локальных переменных не поддерживается — в основном потому, что HHVM достаточно умен, чтобы самому определить их тип. Поэтому наш первый код для проверки в режиме Strict будет выглядеть примерно так:

```
<?hh // strict
  function primefactor(int $num): Vector <int>{
    $sqrt = sqrt($num);
    for ($i = 2; $i <= $sqrt; $i++) {
      if ($num % $i == 0) {
        return primefactor((int)($num/$i))->add($i);
      }
    }
    return Vector {$num};
}
function primeloop(): void {
  print_r(primefactor(pow(2,61)+1));
}
primeloop();
```

Мы явно указываем, что функция **primefactor()** принимает целое число в качестве аргумента (**$num**) и возвращает вектор целых чисел. Обратите внимание, что тип возвращаемого значения указывается после имени функции, а не перед ним, как в C. Мы используем тип **(int)**, чтобы преобразовать результат деления к целому числу, так как уже установили, что **$i** делится на **$num**. Также мы заменили вызов **array_merge()** методом **Vector->add()**.

Если сохранить этот файл под именем **prime.hh** и создать упомянутый ранее файл **.hhconfig**, то при запуске клиента *hh_client* он пожалуется только на одно: на вызов более верхнего уровня **primeloop()**. В режиме Strict код (за исключением выражения **require**) может размещаться только в функциях и классах, что означает: на данный момент невозможно создать (функциональный) проект только на Strict Hack. В коде должен быть хотя бы один файл на Partial Hack, который приводит код в действие. Кроме Strict и Partial, существует режим Decl (декларации), который удобен для подключения существующих проектов на PHP, которые сразу не поддержат подход Hack.

HHVM запускается вместе с web-сервером, таким как *Apache* или *Nginx*, через прокси *FastCGI*. В процессе такого запуска он заменяет обычный интерпретатор PHP, известный как Zend Engine. Предположим, что у вас уже есть один из этих web-серверов, и он настроен на каталог **/var/www**. В составе пакета также есть полезный скрипт, который настроит HHVM и ваш web-сервер на одну страницу. Запустите его следующим образом:

```
$ sudo /usr/share/hhvm/install_fastcgi.sh
```

Затем перезапустите свой web-сервер (мы пользуемся *Apache* на Ubuntu Server) и запустите демон HHVM:

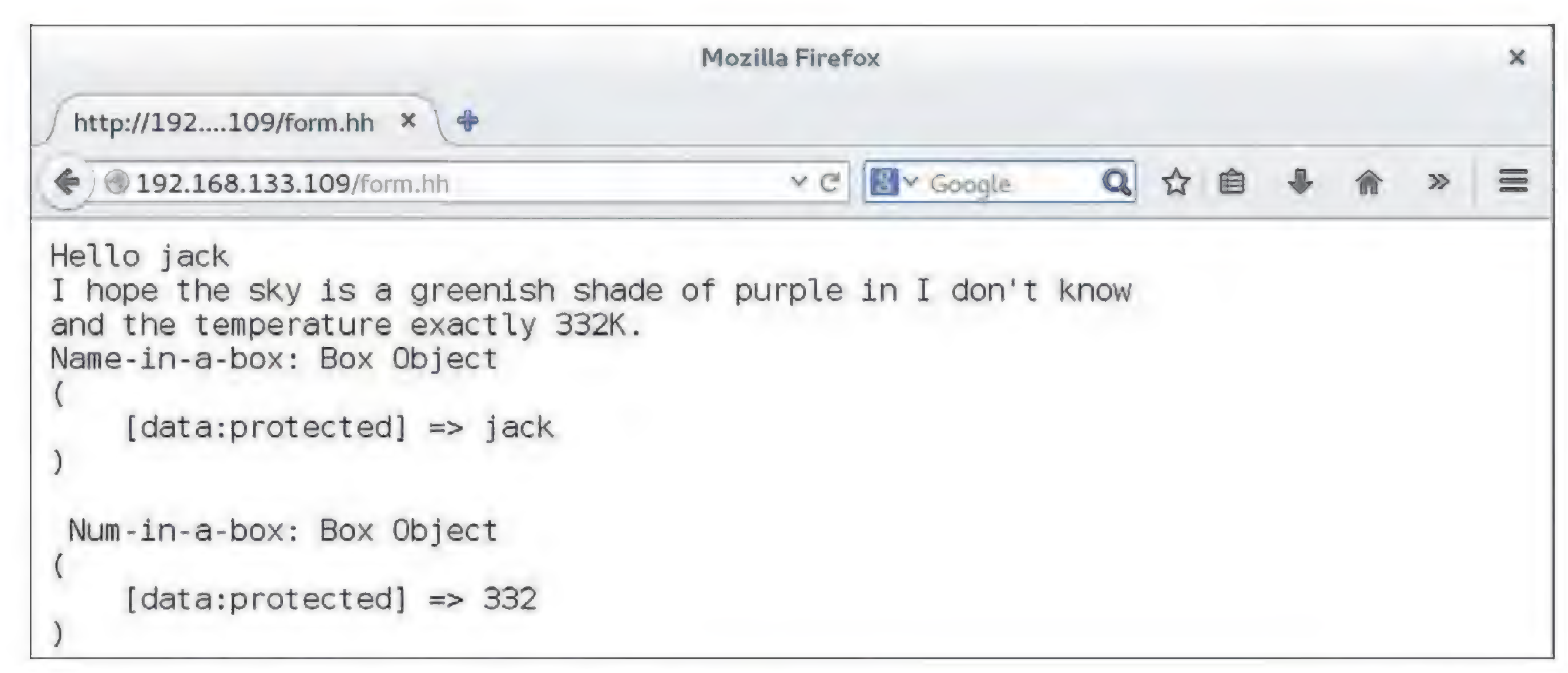
```
$ sudo service apache2 restart
$ sudo service hhvm start
```

В отличие от PHP, код на Hack нельзя вставлять в HTML-файлы. Файлы с HTML-разметкой и кодом должны быть строго отделены друг от друга. Это еще одна составляющая дисциплинированности, которую Hack пытается привить разработчикам. Функция **phpinfo()** для HHVM на данный момент менее подробна по сравнению с PHP: она возвращает только 'HipHop' в отличие от блистательного объема информации своего предшественника (см. рисунок на стр. 97). Если бы мы попытались запустить свой первый код на *Apache*, скорее всего, дело закончилось бы таймаутами. Запускать код, нагружая web-сервер — не самая хорошая идея. Серверу и так забот хватает.

Знакомство с обобщениями

Одна из возможностей зрелого языка, которая появилась в Hack — обобщения. Они позволяют типизировать классы и методы даже тогда, когда их тип может быть неизвестен заранее. Мы используем обобщенный параметр **T** на примере описанного ниже класса **Box**:

```
<?hh
class Box<T> {
  protected T $data;
  public function __construct(T $data) {
    $this->data = $data;
  }
  public function getData(): T {
    return $this->data;
  }
}
```



› **Мы поместили в коробки Джека и число 332.**

Класс Box имеет собственный тип инициализации, и метод **getData** возвращает именно этот тип **T**. Это довольно академичный пример, но код, проверяемый статически, очень удобен для разработчиков.

Формы HTML

Как и PHP, Hack искусно обращается с HTML-формами. Вот очень простая:

```
<html>
<form action="form.hh" method="post">
<p>Как ваше имя? <input type="text" name="name" /> </p>
<p>Какой ваш любимый цвет? <input type="text" name="colour"
/> </p>
<p>Назовите столицу Ассирии: <input type="text"
name="capital" /> </p>
<p>Выберите число с абсолютной величиной, меньшей
2<sup>63</sup> - 1
<input type="text" name="num" /> </p>
<p><input type="submit" /></p>
</form>
</html>
```

В нашем файле **form.hh** для обращения к этим данным используется традиционный массив **$_POST**:

```
<?hh
$name = $_POST['name'];
$colour = $_POST['colour'];
$cap = $_POST['capital'];
$num = (int) $_POST['num'];
echo "<pre>";
echo "Привет, " . $name . "\n";
echo "Надеюсь, что небо " . $colour . « in « . $cap . «\n»;
echo "а температура равна " . $num . "K.\n";
echo "</pre>";
```

Если вы сдуру введете в поле с номером нечисловое значение, то **$num** при преобразовании примет значение 0.

С этим простым примером можно объединить предыдущий академичный пример, и мы увидим, как обобщенный класс **Box** может принимать как строку, так и целое число. Для этого добавим класс **Box** в начало файла **form.hh** и затем припишем следующий код в конец этого файла, заменив им закрывающий тэг </pre>:

```
$boxName = new Box($name);
$boxNum = new Box($num);
echo "Name-in-a-box: ";
print_r($boxName);
echo "\n Num-in-a-box: ";
print_r($boxNum);
echo "</pre>";
```

Таковы вкратце Hack и HHVM, нежданный подарок от Facebook миру... по крайней мере, миру разработчиков PHP. Почему бы не выразить ему свое одобрение, полайкав всякие страницы на http://hhvm.com? LXF

Краткая история хип-хопа

Привет, чувак. Это Нью-Йорк, 1975 год. Ой, не тот хип-хоп. Вернемся лучше к концу предыдущего десятилетия. Тысячи строк PHP в коде Facebook стали перегружать его дата-центры, через которые несколько сотен миллионов пользователей одновременно ставили друг другу лайки/препирались/следили друг за другом. Решением проблемы стал HipHop для PHP (он же HPHPc) — транслятор кода на C++, который компилировался и запускался гораздо быстрее, чем работал громоздкий интерпретатор PHP. Исходный код HPHPc был открыт в 2010 году, но пользовательская база продолжала расти/ставить лайки/следить друг за другом.

HipHop оставил не слишком хорошие результаты, частично потому, что мог компилировать только определенные типы кода PHP, частично из-за растущих затрат на его разработку и поддержку. Возникла необходимость в новом подходе.

Так появилась виртуальная машина HipHop Virtual Machine, которая сначала преобразует код на PHP (а теперь и на Hack) в байт-код HipHop (промежуточный язык, лучше поддающийся анализу и оптимизации), и затем в мгновение ока формируется и выполняется 64-битный машинный код. Все это происходит в соответствии с принципами JIT ('just-in-time'), поэтому этап статической компиляции HPHPc исключается. Это значит, что цикл «изменение–перезагрузка–отладка» вашей очередной программы на PHP проходит гораздо более гладко.

Использование виртуальных машин для кода — определенно не новая идея: виртуальная машина Java (JVM) установлена на нескольких миллиардах устройств, а возможность LLVMpipe архитектуры драйверов Gallium3D обеспечивает гораздо более унифицированный подход к работе с графикой на самых различных видеокартах. Компилятор *Clang* также использует LLVM — и, возможно, в один прекрасный день придет на замену доблестного *GCC*.

LXFHotPicks

Firefox » Rainbow Stream » EtmTk » WP-CLI » Sweet Home » 3D Streamtuner2 » Pyspread » Freeciv » nSnake » UNetbootin » Sedlisp



Ричард Смедли
Ричард бродит по репозиториям проектов и дистрибутивов и просеивает тысячи баз данных, чтобы рассказать вам о лучших релизах месяца.

Web-браузер

Firefox

Версия: 31 **Сайт:** www.mozilla.org/firefox

Спросите себя: «Принимаю ли я *Firefox* как готовую данность?». Возможно, у вас до сих пор версия 26 или 24 (или даже 17). Возможно, вы перестали загружать новые релизы, и даже соглашаться на опцию автоматического обновления? ОК, может, вам и так есть о чем беспокоиться, помимо головокружительной скорости обновления браузеров. Но, переживая по поводу стремления *Chrome* занять место самого популярного браузера — и немного по поводу кадровой политики Mozilla — вы, возможно, кое-что упустили.

Поскольку большинство читателей **LXF** являются конечными пользователями, а некоторые — разработчиками в той или иной форме, вам будет приятно узнать, что *Firefox 31* многое предлагает и тем, и другим. *Firefox* не просто использует новые функции CSS3 и HTML5 (в этом релизе есть каскадные переменные и субтитрирование WebVTT), но и добавляет такие функции, как блокировка вредоносных программ для двоичных закачек.

Безопасность сочетается с удобством использования в новой библиотеке подтверждения сертификатов mozilla::pkix, реализованной 4167 строками кода C++ вместо 81865 строк автоматически переведенного кода (Java в C). Mozilla рассчитывает на обратную связь по вопросам совместимости и безопасности именно в данном конкретном обновлении, поэтому сообщите, пожалуйста, Bugzilla, если обнаружите странности при открытии любимых сайтов.

> «Выводит Firefox в первые ряды дружелюбных к разработчику.»

› Теперь новая вкладка добавляет окно поиска в ярлыки часто посещаемых сайтов.

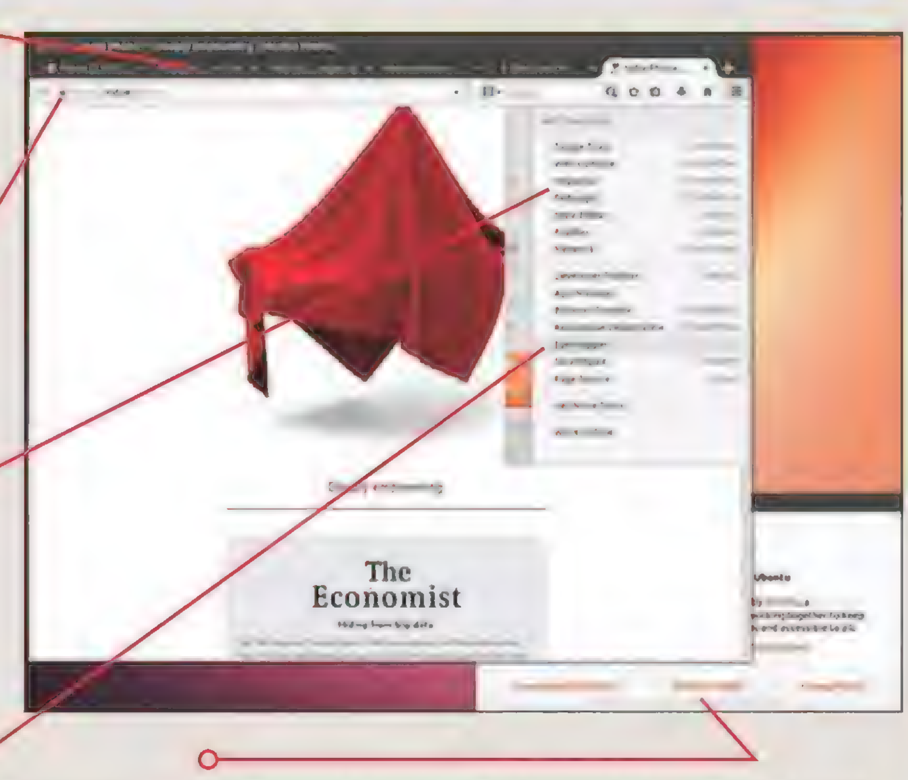
Разработчики

Кстати об ошибках: Add-on Debugger позволяет запускать JavaScript Debugger в контексте дополнения, а новый Canvas Debugger работает с WebGL, так же, как и с 2D Canvas. И это снова выводит *Firefox* в первые ряды программ с дружелюбными к разработчикам функциями.

При выборе Eyedropper в меню Web Developer (см. Исследуем интерфейс *Firefox 31*, слева) курсор примет вид лупы, которая выбирает отдельные пиксели, выделяет интенсивность цвета, позволяет делать копии в буфер обмена и изменять интенсивность цвета в Inspector's Rules. Среди горячих клавиш Eyedropper — Shift+стрелка, чтобы переместить Eyedropper на 10 пикселей (сравните с перемещением по одному стрелкой). Откройте Page Inspector (Ctrl+Shift+C), и здесь вы найдете одну из самых элегантных новых функций — редактирование в режиме live графического вида Box Model.

Firefox 31 также обладает явным преимуществом потребления куда меньшего объема ОЗУ при открытии 80 или 90 вкладок.

Исследуем интерфейс Firefox 31



По-прежнему лучший для тех, кто жить не может без десятков открытых вкладок, загромождающих место.

Этот замочек теперь поддерживает сильная новая библиотека mozilla::pkix, в которой имеется улучшенная перекрестная обработка сертификатов.

Mozilla сражается за аудиторию web-разработчиков, на которую посягает *Chrome*. Сейчас у *Firefox* имеется новый canvas debugger и редактируемая панель Box Model.

Инструмент Eyedropper (в меню Web Developer) предлагает удобную возможность отображения и сохранения шестнадцатеричного цифрового значения из любого пикселя на странице.

Несмотря на всю шумиху по поводу принятия DRM в Mozilla, *Firefox* по-прежнему обращает больше внимания на конфиденциальность, свободу ПО и права конечного пользователя, чем другие браузеры.

Клиент Twitter

Rainbow Stream

Версия: 0.3.10 **Сайт:** www.rainbowstream.org

На настольных ПК собственный web-интерфейс Twitter всегда был популярным и удобным способом взаимодействия с сервисом микроблогов, но он меняется, и не всегда к лучшему. С расширением применения ресурсом JavaScript для тяжелых страниц вроде этой вы, возможно, решите оставить свои вкладки для сервисов, которые не используют дополнительных клиентов.

Неплохая альтернатива— клиент командной строки *Rainbow Stream*. Он хорошо справляется с работой, прост в использовании, нетребователен к ресурсам и не капризничает, как другие клиенты. Установка в Python 2.7 или 3.x довольно проста, разработчики рекомендуют **virtualenv**:

```
virtualenv venv
source venv/bin/activate
pip install rainbowstream
```

Но использование всего лишь

```
sudo pip install rainbowstream
```

тоже неплохо работает, если вам ни к чему полная настройка *virtualenv* в Python.

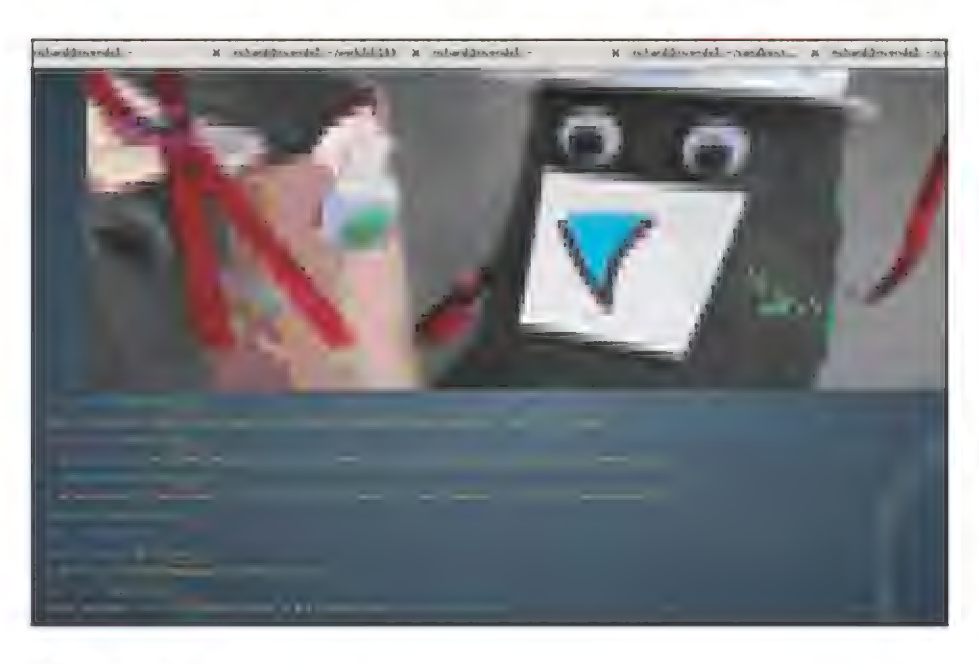
В крайнем случае, вам просто надо будет установить для этого пакет *python-pip* своего дистрибутива вместо *python-virtualenv* (*или python3-virtualenv*).

Дружелюбие к пользователю

Настройка может оказаться приятным сюрпризом для тех пользователей, которые ожидали долгой возни с текстовым файлом. При первом запуске **rainbowstream** ваш браузер по умолчанию откроется и соединится с Twitter, чтобы вы авторизовали соединение клиента с вашей учетной записью, предоставив вам PIN для вставки его в терминале. Вот и все. Мы начали с

```
rainbowstream -iot
```

> «Неплохая альтернатива — клиент CLI Rainbow Stream.»



› Обработка терминала *Rainbow Stream* изображений Twitter добавляет нечто большее к оригиналу.

который превращает изображения, на которые даются ссылки, в ASCII-арт. Нажмите H, чтобы вывести список команд. Размещение не сложнее, чем **t Hello, World!** Номера [ID] после каждого твита облегчают добавление в избранное, пересылку (ретвит) и цитирование отдельных постов.

Для тех из нас, кто по-прежнему интенсивно использует списки Twitter, *Rainbow Stream* предлагает быстрый способ добавить в список и изменить список с клавиатуры.

Одна из проблем для активных пользователей — количество твитов, которое с замечательной скоростью забивают терминал; P и R будут ставить на паузу и снимать с паузы экран, позволяя вам переварить чьи-то 140 символов мудрости.

Календарь и менеджер списка дел

EtmTk

Версия: 3.0.34 **Сайт:** http://bit.ly/ETmTkSource

Пару лет назад **LXF** рассматривал удобный менеджер задач на базе *Qt* под названием *Etm* (event and task manager). Автор, Дэниэл Грэм [Daniel Graham], с тех пор заменил его на более кросс-платформенную версию *EtmTk*, основанную на *Tk* вместо *Qt*.

Обеспечьте наличие трех модулей Python — dateutil, PyYaml и icalendar, под разными вариациями имен в менеджере пакетов вашего дистрибутива, затем распакуйте tar-архив и запустите

```
sudo python setup.py install
```

Ввод **etm ?** даст вам сводку аргументов командной строки, поскольку *EtmTk* прекрасно работает и без GUI. Тем не менее это приложение GUI, и запуск его как такового позволит вводить данные через диалоговые окна, что объединяет традиционный интерфейс WIMP [окно, иконка, меню, курсор] с собственным сжатым синтаксисом *Etm* для настройки записей в календаре или списке важных дел.

Данные сохраняются в виде текстовых файлов в **~/.etm/data/**, что облегчает распределенный доступ с разных машин и, возможно, синхронизацию с мастер-копией на вашем файловом сервере с помощью *Dropbox* или свободной альтернативы с открытым кодом, например, *OwnCloud*.

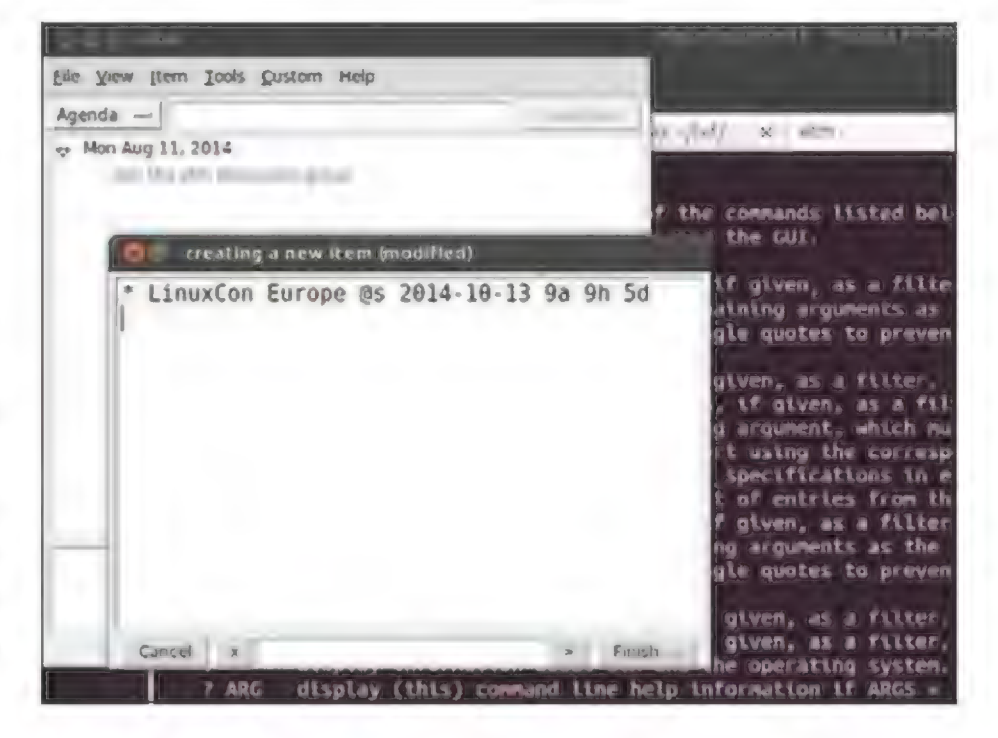
Жесткий синтаксис

Вот пример со стрижкой, взятый из онлайн-руководства и измененный для моего собственного визита к парикмахеру:

```
- get haircut @s 25 @r d &i 21 @o r
```

В данном примере **@s 25** означает, что надо начать с 25-го (текущего месяца), **d &i 21** — временной интервал в 21 день (три недели; за этот срок шевелюра

> «Мы освоились не сразу, но начали учить команды наизусть.»



› Если вы ищете приложение календаря с GUI с возможностью быстрого и простого ввода с клавиатуры, то вот оно.

достаточно отрастает, чтобы начать меня раздражать), а **@o r** дает *Etm* команду повторить после завершения.

Имеется подсказка в формате PDF, занимающая две страницы мелким шрифтом. Мы бы советовали держать ее под рукой: во время нашего краткого опыта использования *EtmTk* она нам очень пригодилась. Мы освоились не сразу, но начали учить некоторые команды наизусть. По мере осознания сжатого синтаксиса, ввод данных станет настолько быстрым, что вы приловчитесь добавлять задачи и события практически со скоростью мыслей о них.

Администрирование сайта

WP-CLI

Версия: 0.16 **Сайт:** http://wp-cli.org

Хотя критиковать *WordPress* легко, его не превзойти в его способности помочь в создании и обработке расширяемого сайта. Единственное, чего ему недостает — это быстрого способа администрирования без помощи браузера. К счастью, этот пробел ликвидирован *WP-CLI*.

На странице GitHub есть двоичные пакеты, позволяющие его установить, или отыщите пакет *wp-cli.phar* и установите его командами

```
chmod +x wp-cli.phar
sudo mv wp-cli.phar /usr/local/bin/wp
```

В качестве альтернативы можете клонировать репозиторий Git, затем запустить скрипт **utils/builddev** для создания символьной ссылки с **/usr/local/bin/wp**.

Подготовив все (понадобятся только PHP и пакет *php-cli*, 5.3.2 или новее, и *WordPress 3.5.2* или более поздний), можете также добавить пакет автозавершения *Bash*. Заметьте, вышеупомянутый скрипт тоже сделает это для вас. Вы можете установить *WP-CLI* на web-сервер без установки *WordPress*, запустив команду

```
wp core download
```

желательно откуда-нибудь в **/var/www**, чтобы скачать *WordPress*.

По вашей команде

WP-CLI поставляется с целым рядом удобных команд, вызываемых по

```
wp plugin install hello-dolly
```

Среди присутствующих команд — **wp comment** для управления комментариями; **wp core** для скачивания, установки, обновления и общего управления самим *WordPress*; **wp help**; **wp media** для управления вложениями; **wp plugin**; **wp post** для управления постами; **wp role** для управления ролями пользователей; **wp theme**; и **wp widget** для управления всеми виджетами боковой панели. Весьма примечательной командой *WP-CLI* является **wp shell**, которая открывает REPL, где вы можете запустить любую функцию *WordPress* для интерактивной оценки.

> «WP-CLI поставляется с целым рядом удобных команд.»



› *WP-CLI* управляет установленным *WordPress*, но позаботится и о скачивании самой свежей версии *WordPress* для вашего нового сайта.

В Справочнике команд [Commands Cookbook] добавление команд в *WP-CLI* описано как «не сложнее, чем создание класса PHP». Как вполне естественное следствие, доступно огромное количество созданных сообществом команд. Загляните в список, имеющийся в wiki проекта на GitHub: https://github.com/wp-cli/wp-cli/wiki.

Приложение для дизайна интерьеров

Sweet Home 3D

Версия: 4.4 **Сайт:** www.sweethome3d.com

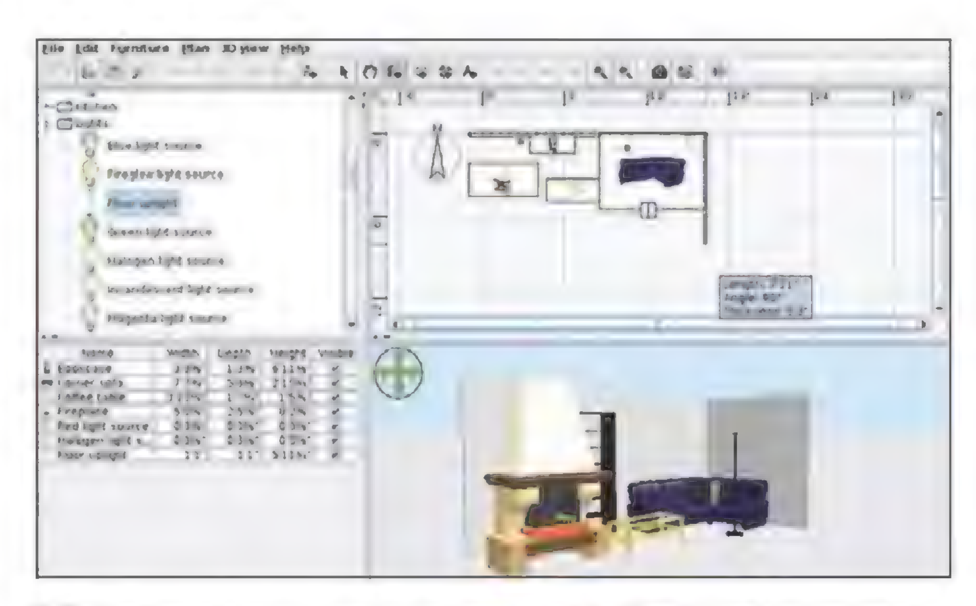
Намерены сделать перестановку в комнате — а то и во всем доме — но не в состоянии объяснить остальным членам семейства, чего именно вы хотите? Вам нужно приложение, которое поможет разместить вашу мебель, а потом покажет вид в 3D. То есть *Sweet Home 3D* от французских Java-разработчиков eTeks.

Установка проста: скачайте и запустите javaws **SweetHome3D.jnlp** с официального сайта. В итоге перед вами появится пустой фон комнаты, с мебелью, которую надо перетаскивать. Точные размеры комнаты для своего плана вы можете ввести позднее. Начните с перетаскивания мебели, чтобы увидеть, как быстро *Sweet Home 3D* обрабатывает вид комнаты. Вы можете как угодно переворачивать вид 3D — здесь нет корявости, обычно присутствующей во множестве приложений Java для GNU/Linux. Однако с большинством указывающих устройств заставить ваши стены и мебель выглядеть должным образом — хитроумная задача.

Создайте что-то свое

Каталог мебели огромен; мы превеселo провели время, устанавливая повсюду винтовые лестницы. Однако он не бесконечен, и вы, вероятно, захотите его пополнить. Есть более 800 бесплатных моделей, разработанных пользователями на сайте скачивания, но можно создать и собственные модели, в *Blender*. *Sweet Home 3D* поддерживает файлы моделей в форматах OBJ, DAE, 3DS и KMZ. В версии 4.4 вы можете редактировать 3D-модель мебели в Furniture Library Editor — это отдельное приложение.

> «Будьте готовы, что члены семьи начнут менять ваш дизайн.»



› Дайте спине отдохнуть: перетаскивать мебель — уж не говоря о камине — куда удобнее мышью.

Подобным же образом можно настраивать цвета и текстуры в комнате, импортировать их и модифицировать с помощью имеющейся библиотеки Textures Library. Вид 3D отлично настраивается, и вы можете сделать фотографию или пошаговое видео всех ваших стараний. Есть даже апплет Java — просмотрщик *Sweet Home 3D*, чтобы вид 3D отображался на вашем сайте.

Построив свой виртуальный дом, экспортируйте план в формат SVG или в вид 3D формата OBJ. Будьте готовы к тому, что другие члены семьи начнут менять ваш дизайн, потому что *Sweet Home 3D* работает также на Mac и Windows.

Браузер интернет-радиостанций

Streamtuner2

Версия: 2.1.1 **Сайт:** http://bit.ly/Streamtuner2

Браузер интернет-радио *Streamtuner2* — это переписанный на Python *Streamtuner* 0.99.99 Жана-Ива Лефора [Jean-Yves Lefort]. *Streamtuner2* легко поддается расширению с помощью плагинов, которые устанавливаются в **/usr/share/streamtuner2/channels/**, оставляя возможности для других сервисов, в том числе и для плагина для работы с MP3 в локальной файловой системе.

При установке Deb вручную, командой

```
dpkg -i streamtuner2-2.1.1.deb
```

мы получили сообщение, что нам не хватает *python-keybinder* и *python-pyquery*. Но

```
apt-get install -f
```

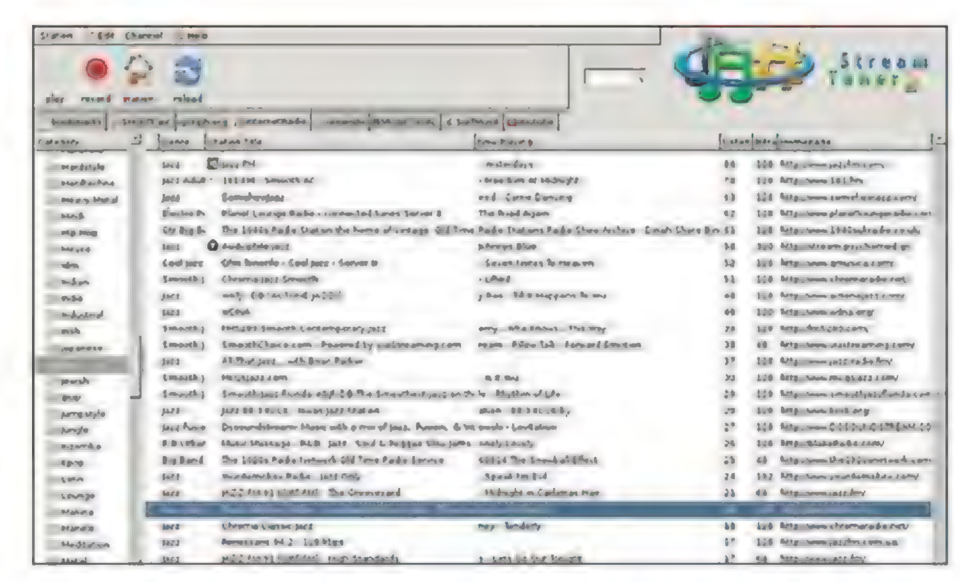
позаботился об этом. Если вы используете более сложный метод установки, чем *dpkg*, например, менеджер пакетов Ubuntu с GUI, то все это будет сделано за один прием. Для других дистрибутивов также имеются пакеты RPM и двоичный установщик.

Мы запустили *Streamtuner2* из командной строки — так всегда стоит поступать с новым приложением — и могли видеть, какие именно сообщения об ошибках появились в терминале. В нашем случае сразу после нажатия на Play появилось'sh:1: audacious: not found', и звука через колонки не было. Мы поняли намек и установили *Audacious*, но могли бы настроить в свойствах и другой плейер (нажав F12). Теперь звук у нас был. Много звуков.

Настройка радио

Огромный выбор интернет-радиостанций — вот что делает столь удобным дружелюбное к пользователю ПО с широкими возможностями поиска, подобное *Streamtuner2*. Собирая такие сервисы, как Shoutcast и Live365 и объединяя их с платформами сообщества, такими, как Xiph и Jamendo, *Streamtuner2* повышает шансы найти что-то новое или заново открыть старых фаворитов. В прекрасных традициях *nix добиваться высокого качества в чем-то одном и давать возможность другим программам делать все остальное, *Streamtuner2* использует *Streamripper* для записи радиостанций и ваш любимый медиа-плейер для воспроизведения найденных станций. *Streamtuner2* — это также отличный способ поиска музыки на YouTube. Обладая возможностью экспортировать списки станций в формате JSON и имея режим консоли для воспроизведения отдельных радиостанций без использования GUI, *Streamtuner2* теперь способен взять на себя заботу обо всех наших интересах в области интернет-радио.

> «Берет на себя заботу о наших интересах в интернет-радио.»



› **Слушайте интернет-радиостанции и ищите музыку на YouTube по отдельным жанрам.**

Электронные таблицы рабочего стола

Pyspread

Версия: 0.3.2 **Сайт:** http://bit.ly/Pyspread

Цель *Pyspread* — стать «самой python'овой программой электронных таблиц». Она заставляет нас поверить в то, что Вселенная — неплохое место; а если вы еще в этом не убеждены, позвольте нам объясниться подробнее.

Pyspread не просто написан на Python: он полагается на выражения Python в самих своих ячейках, которые являются объектами Python и доступ к которым можно получить из других ячеек. Объекты Python могут представлять все, в том числе списки или матрицы, и вызывать любой модуль Python для матричных операций.

Возвращение Jedi

Если репозиторий вашего дистрибутива отстает на несколько релизов, то

```
pip install pyspread
```

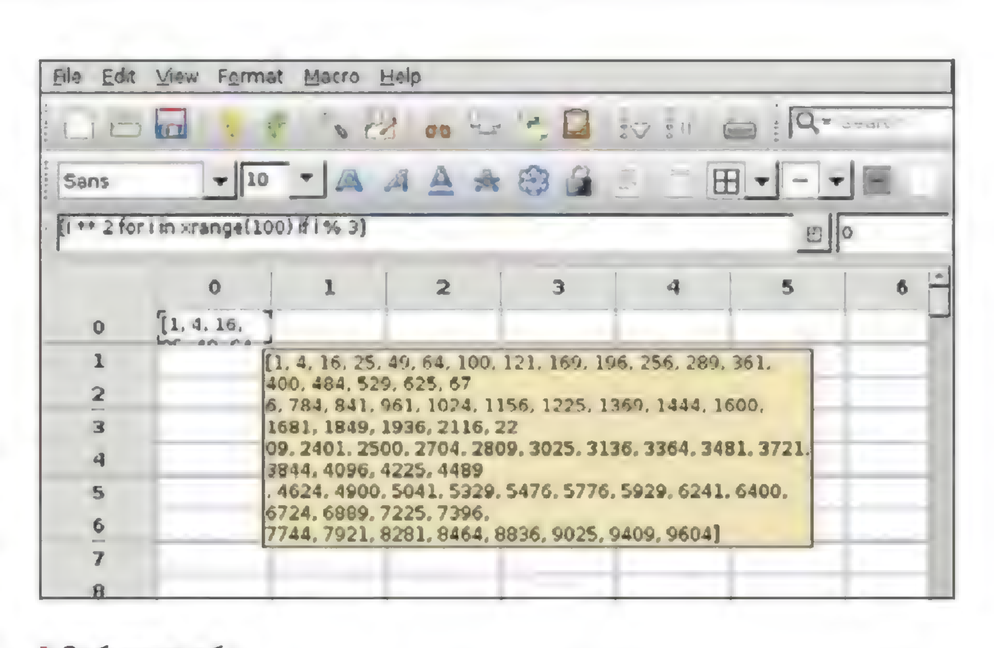
скачает tar-архив и запустит скрипт setup.py — что можно сделать и вручную. Перед этим проверьте, установлены ли у вас *NumPy*, *wxPython* и *matplotlib*. Для приобретения дополнительных функций добавьте *python-gnupg*, который без разрешения откроет ваши личные файлы, или *xlrd* и *xlwt* для загрузки и сохранения файлов *Excel*. Для автодополнения по клавише Tab и контекстной помощи весьма удобен *Jedi*, и есть планшетная карта для образца прогноза погоды в файле PYS.

Pyspread имеет 3D-отображение сетки, в котором он обрабатывает Z (число таблицы) на равных с X и Y (координатами внутри таблицы). S — это объект данных с координатной привязкой, и вы можете ссылаться на отдельные ячейки в формате S[X,Y,Z] для S[строка, столбец, таблица]. Можно добавить диапазон ячеек с помощью sum(S[3:28,5,0]), например.

> «Pyspread слишком мощный для повседневной работы.»

Если вы уже применяете Python в научном контексте, то вы и без нашей подсказки увидите, как хорош *Pyspread*. Для прочих из нас, которые перешли с редактора электронных таблиц *LibreOffice*, *Calc*, это может быть не столь очевидно. На самом деле, *Pyspread*, наверное, слишком мощный для повседневной работы — например, в каждой ячейке может быть функция, которая удалит ваши файлы!

Это выглядит тропой к вредоносному ПО типа MS, но именно поэтому *Pyspread* подписывает все создаваемые им таблицы PGP. Вам также придется утвердить все импортированные файлы, если у вас нет правильного частного ключа. Без него файлы открываются в безопасном режиме и отображается код, но без оценки.



› **Забудьте обо всех этих макросах и спецязыках для электронных таблиц — просто используйте чудный, милый Python для прекрасной работы с данными.**

HotGames Развлекательные приложения

Стратегия

Freeciv

Версия: 2.5.0-beta1 **Сайт:** www.freeciv.org

Лучшее в Hotpicks — что мы не обязаны придерживаться официальных релизов. Вам нужно что-нибудь с пылу, с жару, так что первый бета-релиз — самое оно. Это первый за несколько месяцев *Freeciv*, который нас захватил. *Freeciv* очень похож на *Civilization II* от Сида Мейера [Sid Meier], и разрабатывается уже 19 лет. Он является и онлайн-игрой, и — с усовершенствованием AI — отдельной игрой для ПК. Если хотите в нее сыграть, есть очень функциональная браузерная версия на http://play.freeciv.org.

Играете ли вы онлайн или решили сразиться оффлайн с AI, суть будет одна и та же. Примерно 6000 лет назад вы начинаете создавать цивилизацию. Вы развиваете технологии (начиная с гончарного дела и алфавита, и проходите весь путь до запуска ракет), и по мере своего развития используете ресурсы. Вы расширяете созданную империю, балансируя между дипломатией и войной, и усиленно развиваете свое общество в целом, пока не окажетесь готовы устремиться к звездам.

Захватывающие качества

Есть несколько необходимых условий, но они не особенно экзотические. На нашей тестовой машине мы создали клиент *GTK+ 2.0*. Однако Xaw, *GTK+ 3.0*, SDL и даже *Qt* являются опциями и будут работать безо всяких проблем.

> «Мы с головой окунулись в изометрический мир Freeciv.»



› Можете поставить здесь палатки, но скоро вы изобретете для себя городскую архитектуру.

Данный релиз предлагает в качестве опции новый набор правил civ2civ3. Он кое-что позаимствовал от *Civilization III*, сократив микроуправление и дав шанс новичкам добиться больших результатов.

В 1980-х мы проводили бессчетные часы, играя в настольные игры *Civilization* от Фрэнсиса Трешема [Francis Tresham]. А теперь, решив, что мы избавились от привычки играть в версию Сида Мейера, мы с головой окунулись в изометрический мир *Freeciv*... ладно, до встречи через 6000 лет.

Классическая аркада

nSnake

Версия: 2.0.8 **Сайт:** http://bit.ly/GitHubnSnake

У вас нет свободных 12 часов, чтобы сыграть в *Freeciv*? На другом конце игрового спектра находится эта классическая аркада. Она работает в терминале, так что вам даже не придется снимать руки с клавиатуры.

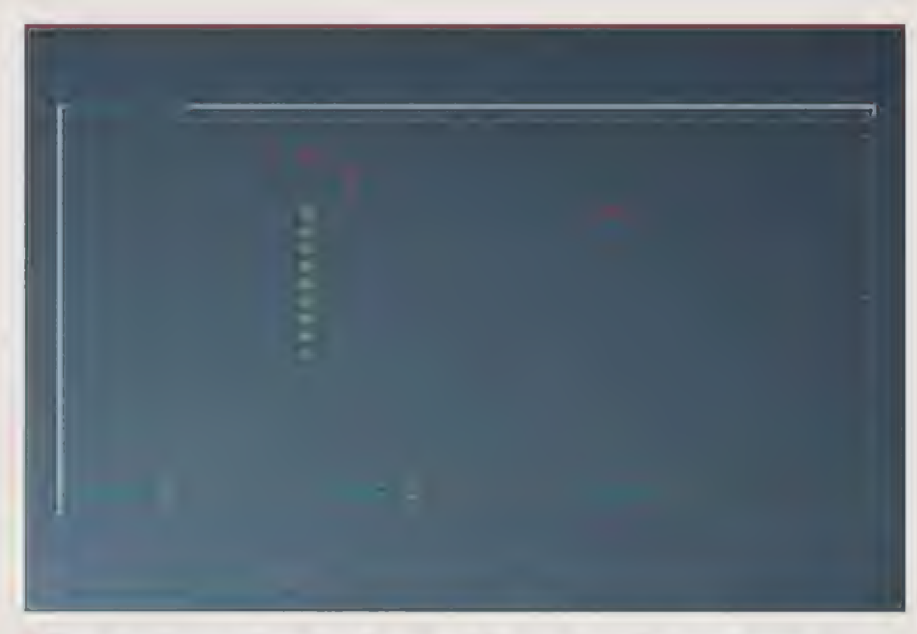
Snake пережил сильный всплеск популярности во время подъема рынка мобильных телефонов, предшествовавших появлению смартфонов, но так приятно видеть новую версию для терминала, способную соперничать с почтенной версией Эрика Реймонда [Eric Raymond] в пакете BSDgames.

В игре вы управляете движущейся змеей, указывая ей направления движения стрелками. Змея питается фруктами. Поедая их, она растет, и постепенно настолько удлиняется, что начинает сталкиваться сама с собой. Игра заканчивается, когда змея натыкается на собственный хвост или на стену.

Страница GitHub проекта утверждает, что *Ncurses* — единственная зависимость, добавляя затем, что нужен пакет *ncurses-dev*. Но, приступив к компиляции (здесь нет шага config, только **make** и **make install**), мы получили несколько ошибок **yaml-cpp/yaml.h**. Поэтому мы добавили в наш дистрибутив пакеты *libyaml-cpp*, и игра установилась без ошибок. Позднее мы прочитали Readme, где говорится, что он «зависит от *Ncurses* и *yaml-cpp*». А сперва мы читали файл **Install.md**, и там ничего подобного не говорилось. Вы усвоили наш опыт? Хм-м.

После компиляции мы сразу вошли в игру. Первый уровень по умолчанию исключительно медленный, что подойдет очень маленьким детям. К счастью, игра неплохо настраивается. Все изменения проходят через меню, и в редактировании текстового файла нет необходимости.

> «10 минут удовольствия, пока ждете компиляции проекта.»



› Если добавить больше фруктов в меню config, придется выполнять чертовски сложные выкрутасы.

Настройте игру, чтобы начать ее с более высокого уровня, добавьте фруктов, случайно расположенных стен и телепортов, и вы получите игру минут на 10 удовольствия, пока вы ждете, когда скомпилируется ваш текущий проект.

Создатель Live USB-диска

UNetbootin

Версия: 608 **Сайт:** http://unetbootin.sf.net

Кросс-платформенная утилита *UNetbootin* используется для создания live USB систем из разных файлов образов дисков. На SourceForge есть бинарник, запускаемый на разных дистрибутивах — просто **chmod u+x** пакет, и создайте программу запуска, если вы не любитель набирать команды:

```
./unetbootin-linux-608.bin
```

Помимо того, что он пишет на флэш-диски USB, *UNetbootin* умеет создавать «скромную программу установки» на вашем жестком диске, которую надо запускать с привилегиями root.

Сила *UNetbootin* в предоставлении вам самых свежих ISO самых удобных дистрибутивов и утилит за один щелчок, избавляющем вас от множества отдельных скачиваний и создания дисков USB. Помимо дюжины популярных дистрибутивов GNU/Linux, вы найдете BSD, в том числе специальные версии типа FreeNAS. Для Ubuntu *UNetbootin* умеет сохранять файлы при перезагрузке образа live с возможностью задания отведенного под эти файлы места. Пользователи Windows могут быстро получить доступ к утилитам восстановления пароля, а диски аварийного восстановления — такие, как Super Grub Disk — позволяют создавать спасательный диск для любой ситуации.

Используя Syslinux, *UNetbootin* делает записи на флэш-диск без форматирования. Это очень удобно, если у вас на диске уже есть некие файлы, и вы просто хотите добавить дистрибутив live, чтобы он путешествовал вместе с ними.



› Большая часть дисков, которые вы хотите создать, всего в паре щелчков от вас.

Lisp, реализованный в sed

Sedlisp

Версия: 20140628 **Сайт:** http://bit.ly/1oVOUJP

Писать собственный компилятор всегда интересно, но реализация Lisp в *sed*...насколько это круто? Чудесное безрассудство Синихиро Хамадзи [Shinichiro Hamaji] является вершиной мастерства в написании *Sed*, которое позволяет использовать полный Lisp на абсолютно любой платформе. Среди встроенных функций — **car**, **cdr**, **cons**, **eq**, **atom**, **+**, **-**, *****, **/**, **mod**, **neg?** и **print**. Среди реализованных Special Forms — **quote**, **if**, **lambda**, **defun** и **define**.

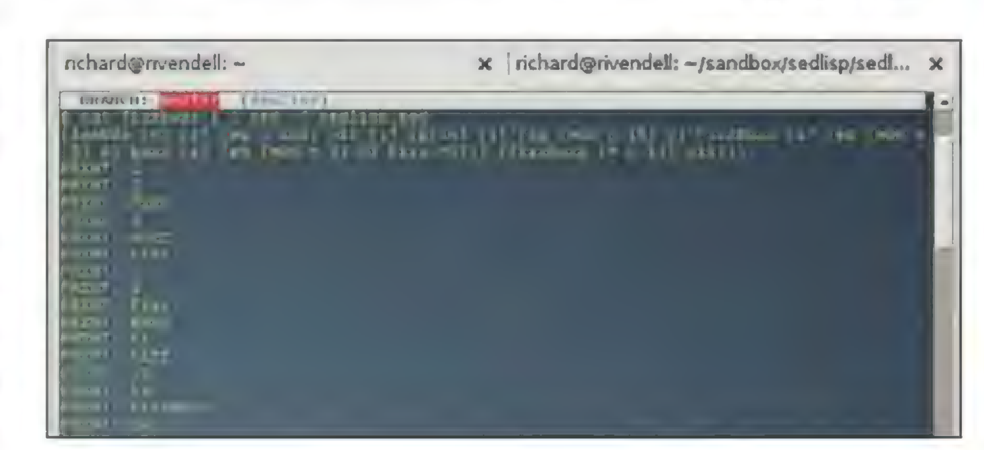
Отдельный eval.l реализует eval и defmacro, завершая удобный, хотя и, по вполне понятным причинам, медленный Lisp. Примеры и тестовый пакет документированы, и показано, как ими надо пользоваться.

Да, по сравнению с другими реализациями Lisp он сильно тормозит, практически застывает, но суть не в этом. Вы когда-нибудь встречали машину без *Sed*? Есть множество машин без Lisp, что грустно, но пока большинство достигнет просветления, вы сможете, по крайней мере, запускать код Lisp на этих компьютерах, передавая исходник через другой текстовый файл в *Sed*, вот так:

```
cat fizzbuzz.l | sed -f sedlisp.sed
```

Lisp обладает истинной мощью в своем минималистском синтаксисе. Иногда вам просто хочется записать свои выражения в Lisp, вне зависимости от того, что еще у вас есть. Я загружаю его на свой диск USB вместе со своей коллекцией интерпретаторов Lisp, написанных на Perl. LXF

› Проведите Lisp туда, где не бывал еще ни один Lisp.



Также вышли

Новые и обновленные программы, тоже достойные внимания...

» Plasma 5.0
Разгон оборудования, high-dpi и «визуально обновленная работа с рабочим столом» для KDE (см. стр. 15).
www.kde.org



› Plasma 5 «визуально обновляет» KDE.

» WordPress 4.0 Beta 2
Обновленный TinyMCE, и ряд усовершенствований для установки плагинов и работы с мультимедийными библиотеками; и много, много отладок.
http://wordpress.org

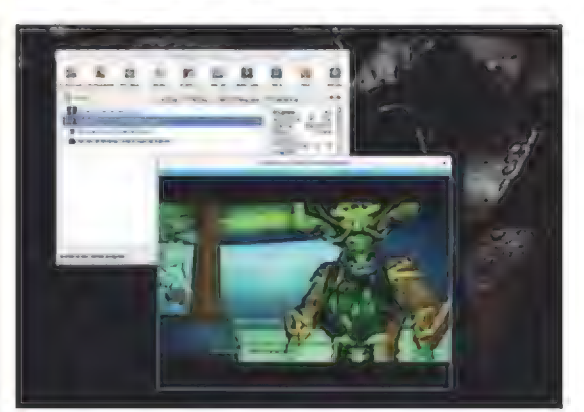
» Micropress 0.2.2
Статический генератор блогов — создает посты с синтаксисом Markdown из командной строки.
https://github.com/glushchenko/micropress

» Firebird 2.5.3
Несколько отладок и разные улучшения в этой достойной реляционной базе данных.
http://www.firebirdsql.org

» Tornado 4.0.0
Масштабируемая web-среда Python и асинхронная сетевая библиотека для FriendFeed.
https://github.com/tornadoweb

» ms-sys 2.4.0
Делает совместимые с Microsoft записи (fdisk/mbr) на жесткий диск или 'sys d:' на флоппи.
http://ms-sys.sf.net

» Wine 1.7.23
Wine получает улучшенную поддержку drag-and-drop файлов и ряд отладок.
http://bit.ly/Wine1723



› Новый релиз *Wine* добавил «начальную поддержку» 64-битной Android.

На диске

Дистрибутивы, приложения, игры, подкасты и всякое-разное...

Лучшее из Интернета, упакованное в 8 ГБ качественного DVD.



Улучшим

Один из дистрибутивов на DVD этого месяца — Deepin; о нем многие из вас и не слыхивали. И хотя есть такие популярные дистрибутивы, как Ubuntu и Fedora — мы полагаем, важно включать и менее известные. Именно так мы поступили десять лет назад (когда балом правили Fedora, SUSE и Mandriva) с производным Debian под названием Ubuntu. Имея странное название и «непохожую на других» коричневую цветовую гамму, он был всего лишь очередным из этих мелких дистрибутивов, которые время от времени появляются и потом тихо исчезают.

И вот теперь, спустя десять лет, мы знаем, что это не тот случай. Ubuntu не только сам стал одним из самых популярных дистрибутивов, но и дал жизнь большему числу производных (включая Deepin), чем любой другой дистрибутив; неплохо для ремикса!

Так где же мы окажемся еще через десять лет? Люди будут продолжать свои жалобы, что Ubuntu опять сменил рабочий стол? Или Ubuntu останется лишь дорогим воспоминанием, как ныне Mandrake, а все вокруг будут использовать Deepin или его производные? Или 2024 наконец-то станет годом Linux на рабочем столе? Понятия не имею, но узнать будет интересно.



» Важно ВНИМАНИЕ!

Порченые диски

В маловероятном случае какого-то дефекта вашего **LXFDVD** обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.

Дистрибутив Linux

Deepin 2014



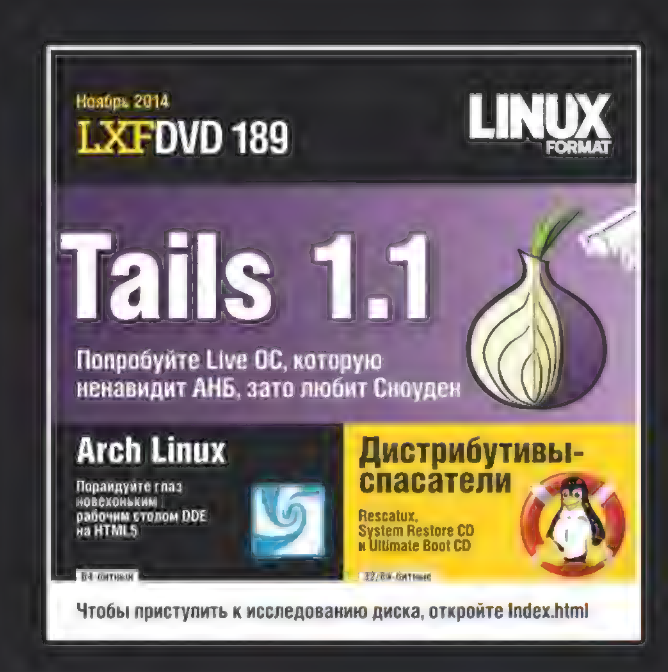
Мы не часто рассказываем о программах из Китая, однако данный дистрибутив привлек наше внимание. Deepin основан на Ubuntu, и тут нет ничего экзотического, но он использует собственную среду рабочего стола, основанную на HTML5. А вот это уже необычно: да, Firefox OS использует HTML5 для своих мобильных приложений, но это целый рабочий стол, причем хороший. И это не просто симпатичный скин для рабочего стола, а полноценная среда с собственными приложениями: плейерами, менеджером файлов и менеджером пакетов/магазином приложений. Добавочное своеобразие Deepin в том, что он использует в качестве браузера по умолчанию *Chrome*, а не *Firefox*.

Дистрибутив Linux

Tails Linux 1.1

Мы уже рассказывали о Tails Linux в **LXF182**, но то была версия 0.22, а теперь он достиг 1.1, так что стоит потратить на него пять минут, верно? И, конечно, в первый раз все было вовсе не ужасно. Tails — в чистом виде live-дистрибутив, цель которого — обеспечить вам максимально возможную конфиденциальность и анонимность в Сети, хотя полностью их гарантировать он не может.

Все соединения идут через сеть Tor для обеспечения анонимности, и он использует лучшие инструменты шифрования с открытым кодом (а только инструментам шифрования с открытым кодом и можно доверять, так как только они делают то, что обещают). Более того, это диск live, и он не оставит никакой информации о том, что вы делали, на компьютере, который вы использовали (если только вы не дадите ему команду сделать это): все куки и кэш хранятся в памяти, и они исчезнут, как только вы перезагрузитесь. Его название расшифровывается как The Amnesiac Incognito Live System (Подверженная амнезии система live инкогнито), что дает весьма правдоподобное описание, хотя эта расшифровка и появилась позже самого названия!



Новичок в Linux? Начните отсюда!

» Что такое Linux? Как его установить?
» Есть ли в нем эквивалент *MS Office*?
» Зачем нужна командная строка?
» Как устанавливать программы?

Ответы приводятся в Index.html на диске.

Дистрибутив Linux

Ultimate Boot CD



Когда имеешь дело с капризным оборудованием, неплохо использовать инструменты диагностики от производителя. То же относится к задачам вроде применения обновлений прошивки. Многие из них предлагаются в виде программ DOS или загрузочных дисков. Ultimate Boot CD сочетает множество таких программ и содержит также специальные инструменты для работы с загрузчиком, разделами и файловыми системами, на одном диске. Загрузитесь в UBCD, затем выберите нужную вам категорию и инструмент. Обратите внимание: поскольку они разработаны как отдельный диск, у вас уже нет безопасного пути в основное меню после того, как вы выберете один из них.

Дистрибутивы Linux

System Rescue CD и Rescatux

Тема данного выпуска — решение проблем, поэтому мы включили пару live-дистрибутивов, разработанных специально для этого. System Rescue CD имеет практически все необходимое для восстановления поврежденной системы или системы, работающей со сбоями, плюс парочку инструментов для работы с Windows. Он предлагает и 32-, и 64-битную версию; 64-битные компьютеры вполне способны работать с 32-битными дистрибутивами, но для того, чтобы справиться с некоторыми проблемами, нужно быть в родной среде. Помимо того, что он крошечный, в нем есть множество опций загрузки, однако основной выбор заключается в том, загрузиться ли вам в текстовую консоль или в рабочий стол *Xfce*. Большинство проблем решается из консоли, но бывает удобно иметь под рукой браузер, когда ждешь помощи от других. С помощью System Rescue CD можно сделать практически все; главное — знать, что следует делать.



Еще один наш дистрибутив аварийного восстановления, Rescatux, предлагает другой подход, в виде своего окна Rescapp, снабженного кнопками для работы с основными проблемами.

И еще!

Системные инструменты

Главное

Checkinstall Установка tar-архива с помощью менеджера пакетов.

GNU Core Utils Основные утилиты, обязанные быть в каждой операционной системе.

Hardinfo Инструмент для тестирования системы.

Ядро Исходник последней версии ядра.

Memtest86+ Проверьте свое ОЗУ на сбои и ошибки.

Plop Простой менеджер загрузки для запуска операционных систем.

RaWrite Создавайте загрузочные дискеты в Windows.

SBM Независимый от ОС менеджер загрузки с простым в использовании интерфейсом.

WvDial Соединяйтесь с Интернетом через телефонный модем.

Чтение

Книжная полка

The Cathedral and the Bazaar [Собор и Базар] Классический текст Эрика С. Реймонда [Eric S Raymond], объясняющий преимущества открытой разработки.

Linux Kernel in a Nutshell [Ядро Linux в двух словах] Введение в ядро, написанное магистром хакеров Грегом Кроа-Хартманом [Greg Kroah-Hartman].

Справочник администратора Debian Руководство для системных администраторов.

Словарь Linux Linux от А до Z.

Окунитесь в Python Мастер-класс по этому популярному языку.

Руководство по созданию скриптов Bourne Shell Начинайте осваивать скрипты оболочки.

Расширенное руководство по созданию скриптов Bash Больше информации для опытных пользователей.

Intro to Linux Руководство для новичков.

Руководство сисадмина Основы работы с системой.

Обзор утилит Обзор инструментария GNU.

Информация о диске

Внимательно прочтите это перед тем, как использовать LXFDVD!

Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials [Главное]» на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

» **имя_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;

» **имя_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;

» **имя_программы-1.0.1.tar.gz** — обычно это исходный код;

» **имя_программы-1.0.1.tgz** — тот же файл, что и выше этажом по списку: "tgz" — это сокращение от "tar.gz";

» **имя_программы-1.0.1.tar.bz2** — тот же файл, но сжатый bzip2 вместо обычного gzip;

» **имя_программы-1.0.1.src.rpm** — также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;

» **имя_программы-1.0.1.i386.FC4.RPM** — двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;

» **имя_программы-1.0.1.ppc.Suse9.rpm** — двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;

» **имя_программы-devel-1.0.1.i386.rpm** — версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Ноябрь 2014
LXFDVD 189
LINUX FORMAT

Slacko
Puppy Slacko 6.0
ISO-образ Live с установкой
32-битный



Robolinux Gnome 7.7.7
ISO-образ Live
64-битный

Ноябрь 2014
LXFDVD 189
LINUX FORMAT

Tails 1.1
Попробуйте Live ОС, которую ненавидит АНБ, зато любит Сноуден



Arch Linux
Порадуйте глаз новехонькими рабочим столом DDE на HTML5
64-битный



Дистрибутивы-спасатели
Rescatux, System Restore CD и Ultimate Boot CD
32/64-битные



Чтобы приступить к исследованию диска, откройте index.html

Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.

DVD-приложение к журналу Linux Format № 11 (189). Ноябрь 2014

» Содержание

LINUX FORMAT

Сторона 1

Сторона 2

Пожалуйста, перед использованием данного диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109!

КОММЕНТАРИИ? Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДЕФЕКТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь, пожалуйста, по адресу disks@linuxformat.ru

Тираж изготовлен ООО «Уральский электронный завод». 620137, Россия, г. Екатеринбург, Студенческая ул., д. 9. Лицензия МПТР ВАФ 77-15.



В декабрьском номере

Главные фавориты 2014

Разбираемся с породой и природой лучших дистрибутивов Linux — почему они быстрее, элегантнее и вообще лучше.

Альтернативы Raspberry Pi

Время паять: Pi мы любим, но не станет ли сердцем вашего проекта другая плата?

Ubuntu — 10!

Мы нацепили бумажные колпаки и чествуем десятилетие Ubuntu, вспоминая его тернистый путь.

Сравнение браузеров

Надоели *Firefox* и *Chrome*? Вот вам лучшие из остальных.

Содержание будущих выпусков может меняться — вдруг мы не успеем дотестировать дистрибутивы... 300 штук осталось...



Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФС77-21973 от 14 сентября 2005 года.
Выходит ежемесячно. Тираж 3000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ
Главный редактор
Кирилл Степанов info@linuxformat.ru
Литературный и выпускающий редактор
Елена Толстякова
Переводчики
Елена Ессяк, Светлана Кривошеина, Валентин Развозжаев, Елена Толстякова
Редактор диска
Александр Баракин
Верстка, допечатная подготовка
Сергей Рогожников
Технический директор
Андрей Смирнов
Директор по рекламе
Владимир Савельев advert@linuxformat.ru
Генеральный директор
Павел Фролов
Учредители
Частные лица
Издатель
ЗАО «Мезон.Ру»

Отпечатано в типографии ООО «ЛД-ПРИНТ»
196644, Санкт-Петербург, Колпинский р-н, пос. Саперный, территория предприятия «Балтика», д. б/н, лит. Ф
Тел. (812) 462-8383, e-mail: office@ldprint.ru
Заказ 11659

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ
Редактор Нейл Мор [Neil Mohr] neil.mohr@futurenet.com
Заместитель редактора Мэтью Хэнсон [Matthew Hanson] matthew.hanson@futurenet.com
Научный редактор Джонни Бидвелл [Jonni Bidwell] jonni.bidwell@futurenet.com
Выпускающий редактор Крис Торнетт [Chris Thornett] chris.thornett@futurenet.com
Художественный редактор Эфраин Эрнандес-Мендоса [Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

ПОДГОТОВКА МАТЕРИАЛОВ
Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Джолион Браун [Jolyon Brown], Марко Фиоретти [Marco Fioretti], Мэтт Хэнсон [Matt Hanson], Дэвид Хейвард [David Hayward], Хью Лэнгли [Hugh Langley], Эндрю Моллетт [Andrew Mallett], Лес Паундер [Les Pounder], Лили Прасьютсут [Lily Prasuethsut], Шивон Рэмси [Siobhan Ramsey], Маянк Шарма [Mayank Sharma], Ричард Смедли [Richard Smedley], Алан Стивенс [Alan Stevens], Генри Винчестер [Henry Winchester], Евгений Балдин, Михаил Остапкевич, Алексей Федорчук, Сергей Яремчук

Ассистенты по выпуску Элис Липскомб-Саутвелл [Alice Lipscombe-Southwell], Эстер Ньюмен [Esther Newman], Алекс Саммерсби [Alex Summersby]
Художественный ассистент Сара Кларк [Sarah Clark]
Иллюстрации Шейн Коллиндж [Shane Collinge], Элли Уоттон [Elly Watton], iStock Photo

КОНТАКТНАЯ ИНФОРМАЦИЯ
UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Тел. +44 01225 442244, email: linuxformat@futurenet.com
РОССИЯ:
Санкт-Петербург (редакция):
пр. Медиков, 5, корп. 7. Тел. +7 (812) 309-0686
По вопросам сотрудничества, партнерства, оптовых закупок: partner@linuxcenter.ru



BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

Новое поколение средств защиты

Межсетевые экраны ССПТ, не имеющие IP-адреса

ССПТ-2 — это сертифицированное ФСТЭК, ФСБ и ГАЗПРОМСЕРТ средство защиты информации нового поколения, реализующее функции межсетевого экрана, но при этом остающееся «невидимым» для любых протоколов и тестовых воздействий, что достигается за счет отсутствия физических и логических адресов на его фильтрующих интерфейсах. **ССПТ-2 невозможно обнаружить никакими известными средствами удаленного мониторинга сети.**

Скрытность функционирования межсетевого экрана повышает надежность системы защиты в целом и существенно упрощает процедуру установки ССПТ-2 в компьютерные сети и функционирующие на их основе информационные и телематические системы.

Защита для высокоскоростных корпоративных сетей Ethernet 100/1000 Мбит/с

Сертифицированы ФСТЭК и ФСБ (3-й класс защиты)

На базе процессоров с 64-разрядной многоядерной архитектурой



Назначение устройства

Основное средство защиты для реализации различных политик информационной безопасности с помощью:

- фильтрации пакетов на канальном, сетевом, транспортном и прикладном уровнях;
- управления транспортными соединениями между отдельными узлами ЛВС или виртуальной ЛВС (VLAN);
- контроля контента данных на прикладном уровне с учетом направления, времени и типа протоколов передачи трафика.

Дополнительное устройство защиты для:

- обеспечения безопасности функционирования ранее установленных в компьютерной сети средств защиты и устройств маршрутизации;
- мониторинга трафика с возможностью анализа данных регистрации пакетов по различным критериям и интеграции с IDS;
- обеспечения функционирования сетевых распределенных телематических приложений и GRID-ресурсов.





